

D1.4 Externí rozhraní CS OTE

Formáty zpráv pro nový VDT/VT trh

Projekt číslo: 00074102

Dokument č.: D1.4.2

Verze dok.: E

Datum vydání: 10.11.2016

OBSAH

1. Úvod	5
2. Popis změn v externích rozhraních.....	6
2.1. Komunikační protokol.....	6
2.2. Připojení na MQ server	6
2.3. Typy výměny zpráv	7
2.3.1. Request-Response komunikace	7
2.3.2. Hromadné zprávy – Broadcast	8
2.3.3. Distribuční pravidla	8
2.3.4. Použití sekvencí pro Broadcast zprávy.....	9
2.3.5. Neplatné a neroutovatelné dotazy.....	9
2.3.6. Zpracování při výpadku.....	9
2.4. Komunikační scénáře	10
2.4.1. Přihlášení, odhlášení uživatele	10
2.4.2. Práce s nabídkami.....	10
2.4.3. Dotaz na veřejná data nabídek.....	12
2.4.4. Dotaz na veřejná data obchodů.....	13
2.4.5. Dotaz na Informativní zprávy.....	13
2.4.6. Dotazy na Produkty a Kontrakty trhu.....	14
2.4.7. Dotaz na Stav trhu	15
2.5. Komunikační zprávy	16
2.5.1. Obecné informace.....	16
2.5.2. Obecné dotazy a odpovědi.....	18
2.5.3. Zavedení a správa nabídek	21
2.5.4. Informace o trhu	26
2.6. Nové scénáře pro stávající způsob automatické komunikace přes komunikační server KSP/KSM .	35
2.6.1. Nastavení/změna/odpověď k novému offline limitu	35
2.6.2. Zpráva o přesunu části offline limitu do online.....	36
3. Použití elektronického podpisu	37
3.1. Příklad zprávy s použitím elektronického podpisu.....	37
4. Šablony XSD	38

Seznam obrázků

Obrázek 1 – Komunikace s MQ serverem.....	6
Obrázek 2: Připojení k MQ serveru a architektura toku zpráv	7
Obrázek 3 – Sekvenční schéma přihlášení/odhlášení uživatele	10
Obrázek 4 – Sekvenční schéma zavedení nabídky s jejím zobchodováním a modifikace nabídky bez vzniku obchodu	11
Obrázek 5 – Sekvenční schéma zadání neúspěšné nabídky	11
Obrázek 6 – Sekvenční schéma hromadné modifikace nabídek (deaktivace) a následný dotaz na nabídky.....	12
Obrázek 7 – Sekvenční schéma provedení dotazu na nabídky	13
Obrázek 8 – Sekvenční schéma provedení dotazu na obchody.....	13
Obrázek 9 – Sekvenční schéma provedení dotazu na Zprávy trhu.....	14
Obrázek 10 – Sekvenční schéma provedení dotazu na Produkty a Kontrakty	15
Obrázek 11 – Sekvenční schéma provedení dotazu na Stav trhu	15

Historie změn History of changes

<i>Datum</i>	<i>Předmět</i>	<i>Revize</i>
13.5.2016	Finalizace obsahu	B
9.8. 2016	Upřesnění směřování odpovědí v detailu formátu zpráv	C
4.10.2016	Rozšíření zpráv pro Blokové nabídky. Upravené zprávy: OrdrEntry, OrdrModify, OrdrExeRprt, PblcOrdrBooksReq, PblcOrdrBooksResp, ContractInfoRprt, ProdInfoRprt. Aktuální verze zpráv změněna na hodnotu 2.	D
10.11.2016	Upřesnění významu atributu Duration zprávy ContractInfoRprt pro zohlednění blokových kontraktů.	E

1. ÚVOD

Cílem tohoto dokumentu je poskytnout popis nového rozhraní pro VDT/VT trh prostřednictvím AMQP serveru.

Pokud budou externí účastníci používat OTE klientskou aplikaci, ta v sobě již toto rozhraní a komunikaci zahrnuje. Pokud však bude externí účastník vyžadovat napojení nového OTE VDT/VT na své systémy, pak by mu měl tento dokument poskytnout popis nutných změn v rozhraní pro implementaci.

2. POPIS ZMĚN V EXTERNÍCH ROZHRAŇÍCH

Z důvodu zajištění vysoké propustnosti a rychlé distribuce zpráv z trhu VDT/VT se rozšiřuje CS OTE o další platformu podporující AMQP protokol. AK v oblasti těchto trhů bude již výhradně probíhat prostřednictvím komunikace s AMQP serverem RabbitMQ. Oproti současnému řešení AK nebude vyžadováno zvláštní nastavení/povolání na straně OTE. Rozhraní pro AMQP server RabbitMQ bude všem účastníkům k dispozici bez rozlišení klienta (identifikace prostřednictvím certifikátu)

Účastník si musí provést implementaci svého klienta, který se bude k MQ serveru připojovat a prostřednictvím kterého bude posílat své požadavky a přijímat odpovědi a hromadné zprávy. Je možné využít AMQP klientskou knihovnu RabbitMQ – viz. webové stránky produktu www.rabbitmq.com.

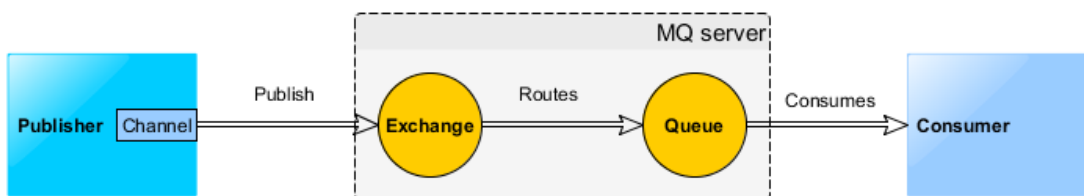
Postup navázání komunikace a jednotlivé komunikační scénáře jsou popsány v následujících částech.

2.1. Komunikační protokol

Komunikace s MQ serverem probíhá prostřednictvím protokolu AMQP (Advanced Message Queuing Protocol). Jedná se o otevřený standard pro komunikační vrstvu aplikací pracujících na datové výměně pomocí zpráv. Implementace bude provedena prostřednictvím MQ serveru RabbitMQ verze 3.6.x.

AMQP standard definuje základní entity:

- Exchange – vstupní bod pro příjem zprávy
- Routes – směrování (distribuce) zprávy
- Queue – Výstupní fronta zpráv



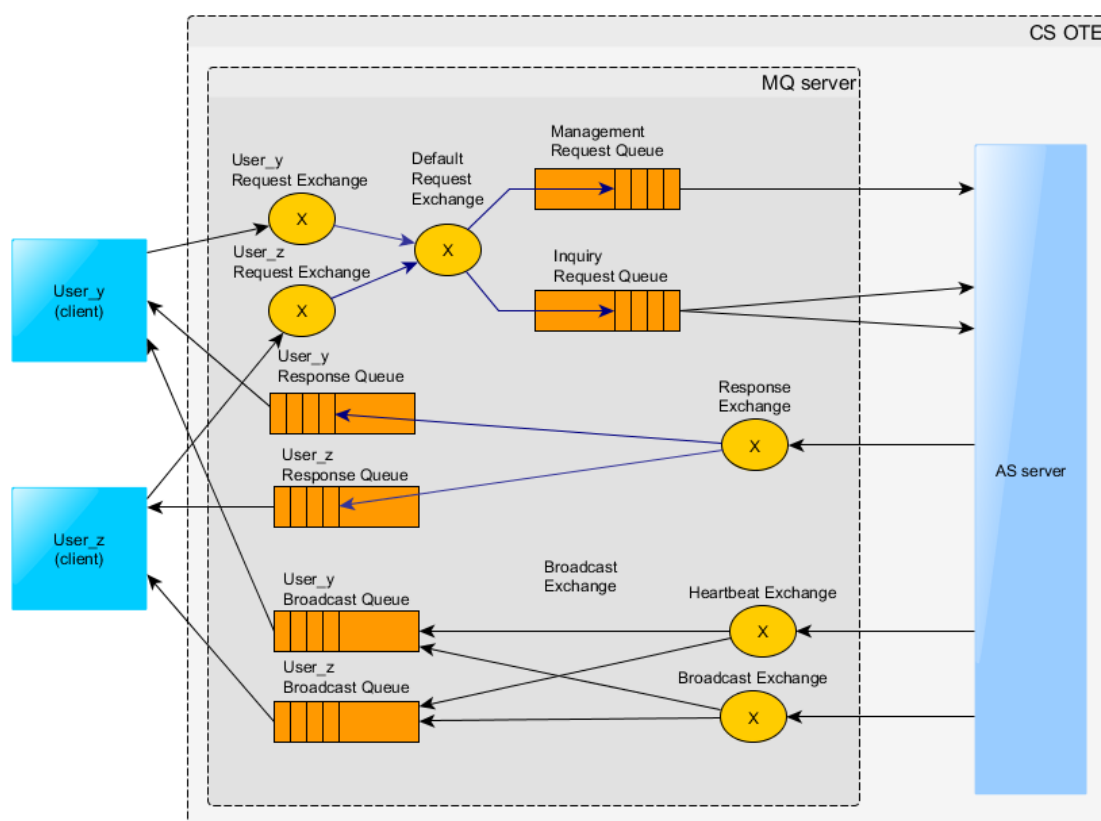
Obrázek 1 – Komunikace s MQ serverem

2.2. Připojení na MQ server

Pro připojení jsou externímu účastníkovi poskytnuté následující technické informace: adresa RabbitMQ serveru, port a identifikace virtual host (viz. dokumentace pro připojení k RabbitMQ <http://www.rabbitmq.com/api-guide.html>). Externí účastník poskytuje OTE svůj klientský certifikát.

Prvním krokem je navázání spojení „*connection*“ na MQ server. Pro vytvoření „*connection*“ je zapotřebí klientský certifikát účastníka, který je nutno nejprve zaregistrovat v systému OTE

Na základě tohoto spojení jsou vytvářeny komunikační kanály „*channels*“, které se připojují k jednotlivým „*queue*“, které slouží pro vzájemnou komunikaci mezi klientem a serverem.



Obrázek 2: Připojení k MQ serveru a architektura toku zpráv

2.3. Typy výměny zpráv

Pro komunikaci Klient – MQ server jsou použité dva základní typy komunikace:

- Dotaz-odpověď (request-response) – dotazy nebo požadavky iniciované klientem, na které MQ server asynchronně odpoví. Odpověď je odeslána pouze iniciátorovi komunikace.

Hromadná zpráva (broadcast) – plošné rozesílání zpráv z MQ serveru na klienty. Rozesílání se provádí na základě definovaných distribučních pravidel a přístupových práv.

2.3.1. Request-Response komunikace

Každý uživatel má na RabbitMQ serveru vytvořenou svou privátní „Exchange“ s názvem „market.exchanges.clientRequest.[USER_ID]“, která slouží pro zadávání požadavků od klienta k MQ serveru. Právo zápisu do této konkrétní exchange má pouze daný uživatel.

Pro příjem privátních zpráv používá uživatel frontu nazývanou „response queue“, která není předvytvořena na AMQP serveru, ale je vytvářena jednotlivými klienty. Při startu aplikace si musí klient vytvořit svou anonymní frontu s automaticky generovaným názvem, jehož jméno je dále použito v elementu *reply-to* ve všech zprávách.

Typy požadavků:

- Pokyn (Management request) – zadání nabídky, modifikace, anulace Dotaz (Inquiry request) – dotaz na obchodní data

Při zadání požadavku typu „Pokyn“ je uživateli vrácena okamžitě odpověď zprávou „AckResp“ Tabulka 7 – Struktura zprávy Logout Report

Acknowledgement **Response (AckResp)** (distribučována do ResponseQueue) a po zpracování požadavku v systému je odeslána odpovídající odpověď na zadaný pokyn (distribučována do BroadcastQueue). Pokud daný pokyn způsobí změnu v obchodních datech, je odeslána hromadná zpráva na všechny uživatele, kterých se změna týká, s odpovídajícím obsahem.

Při zadání požadavku typu „Dotaz“ je uživateli odeslána příslušná odpověď do jeho privátní fronty pro odpovědi (ResponseQueue).

2.3.2. Hromadné zprávy – Broadcast

Systém poskytuje 2 základní typy hromadných zpráv

Market data zprávy – zprávy o změně v obchodních datech a změně stavu trhu. Zprávy jsou distribuované na všechny přihlášené uživatele s příslušnými oprávněními na dané trhy. Heartbeat zprávy – zprávy pro ověření aktivního spojení s klientem. Každý uživatel má na RabbitMQ serveru vytvořenou svou privátní frontu zpráv s názvem „market.broadcastQueue.[USER_ID]“, ke které je připojen a dané zprávy vybírá. Pokud nebude uživatel zprávy průběžně vybírat, může dojít k zahlcení jeho fronty a nové zprávy již nebudou do jeho fronty zařazované. Tím se vystavuje riziku, že neobdrží všechny informace z trhu.

2.3.3. Distribuční pravidla

Popis distribučních pravidel uvádí následující tabulka. Některé klíče jsou dynamicky definované podle aktuálního nastavení trhu a přístupových práv uživatele.

Distribuční klíč	Popis
public	veřejná informace, distribuována na všechny uživatele
public.<marketId>	veřejné informace o daném trhu, distribuována na všechny uživatele, kteří mají přístup na daný trh
public.trade.<prodName>	veřejné informace o obchodu, distribuována na všechny uživatele, kteří mají přístup na daný produkt
PRTC_<particId>	informace relevantní pro konkrétního účastníka trhu
<prodName>	informace relevantní pro produkt
<prodName>. PRTC_<particId>	informace relevantní pouze pro PARTIC_ID vzhledem k produktu
trade	informace o obchodech pouze pro administrátory (obsahují obě strany obchodu)
halfTrade.<prodName>. PRTC_<particId>	privátní informace o vzniklých obchodech (obsahuje pouze polovinu obchodu pro daného účastníka)
USR_<userId>	privátní informace určené pouze pro daného uživatele

Tabulka 1 – Přehled distribučních pravidel

Pro názornost je uveden následující příklad na konkrétním uživateli.

Uživatel: 123, Účastník: 12, Přístup na trh: INTRADAY, Dostupné produkty: INTRADAY_1H

Uživatel bude dostávat zprávy, které budou zaslány s některým z následujících distribučních klíčů:

- public
- public.INTRADAY
- public.trade.INTRADAY_1H
- PRTC_12
- INTRADAY_1H
- INTRADAY_1H.PRTC_12
- halfTrade.INTRADAY_1H.PRTC_12
- USR_123

2.3.4. Použití sekvencí pro Broadcast zprávy

Sekvenční čísla jsou použité pro identifikaci pořadí Broadcast zpráv jestli nějaká zpráva nebyla ztracena. Sekvenční číslo není přímo obsaženo v těle zprávy, ale je součástí hlavičky AMQP zprávy jako atribut „market-group-sequence“.

Sekvence je vždy navýšena o jedna pro každou další Broadcast zprávu. Sekvence jsou uchovávány pouze v paměti (nejsou ukládány), což znamená, že při restartu nebo ukončení serveru bude nastavena na hodnotu 0. Pokud klient obdrží neočekávanou hodnotu (hodnota je jiná než poslední číslo + 1), měl by si vyžádat aktuální data trhu z systému CS OTE.

Sekvenční čísla jsou počítané pro jednotlivé distribuční klíče (routing keys - atribut „market-group-id“ v záhlaví zprávy). Pro jednotlivé distribuční seznamy budou tedy čísla sekvencí odlišná. Fronty připojené na default broadcast exchange s stejným distribučním klíčem obdrží totožné číslo sekvence.

2.3.5. Neplatné a neroutovatelné dotazy

Pokud CS OTE systém nemůže zpracovat požadavek z důvodu, že požadavek není koretní nebo nemůže být zpracován, je odeslána negativní odpověď. Odpověď obsahuje detail s uvedením důvodu, proč požadavek nebyl zpracován.

Pokud CS OTE systém nemůže zpracovat dotaz z důvodu neplatné nebo chybějící verze v hlavičce zprávy, je odeslána nativní error odpověď. Odpověď má nastavený atribut content-type na hodnotu `market/error`. Tělo zprávy obsahuje chybovou zprávu kódovanou v UTF-8. Odeslání nativní chybové odpovědi je vyvoláno zjištěnou chybou při validaci zprávy systémem CS OTE. Tyto validační chyby mohou nastat v následujících případech:

- Neplatné XML schéma
- Nevyplněn atribut User ID
- Nevyplněn atribut ContentType
- Nevyplněn atribut ReplyTo
- Nevyplněn atribut CorrelationId

Pokud CS OTE systém nemůže zpracovat požadavek, protože neběží, požadavek bude zrušen na straně AMQP serveru a klient o této situaci bude informován prostřednictvím jeho „return listener“.

2.3.6. Zpracování při výpadku

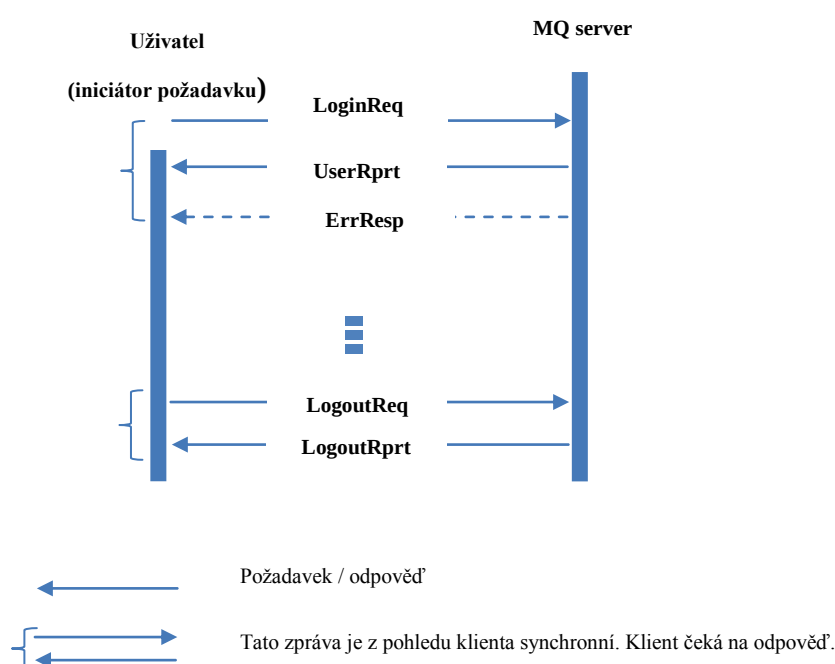
V případě, že AMQP server neběží (z důvodu výpadku nebo restartu), klientské přihlášení je ztraceno. Pokud má klient registrován „shutdown listener“, obdrží o výpadku notifikaci z AMQP serveru. Po úspěšném opětovném připojení na AMQP server se klient musí opět přihlásit.

2.4. Komunikační scénáře

2.4.1. Přihlášení, odhlášení uživatele

Základní komunikační scénář pro přihlášení, odhlášení uživatele k systému a požadavek na aktuální informace o systému. Uživatel po navázání spojení s MQ serverem, musí zahájit datovou komunikaci prostřednictvím požadavku na přihlášení *LoginReq*. Při úspěšném ověření je odpověď zpráva *UserRprt*, v případě neúspěchu je odeslána zpráva na klienta *ErrResp*.

Při ukončování klientské aplikace, uživatel je povinen poslat zprávu odhlášení *LogoutReq*. Pokud uživatel nepošle požadavek na odhlášení, bude automaticky odhlášen dle definovaných pravidel při ztrátě spojení.



Obrázek 3 – Sekvenční schéma přihlášení/odhlášení uživatele

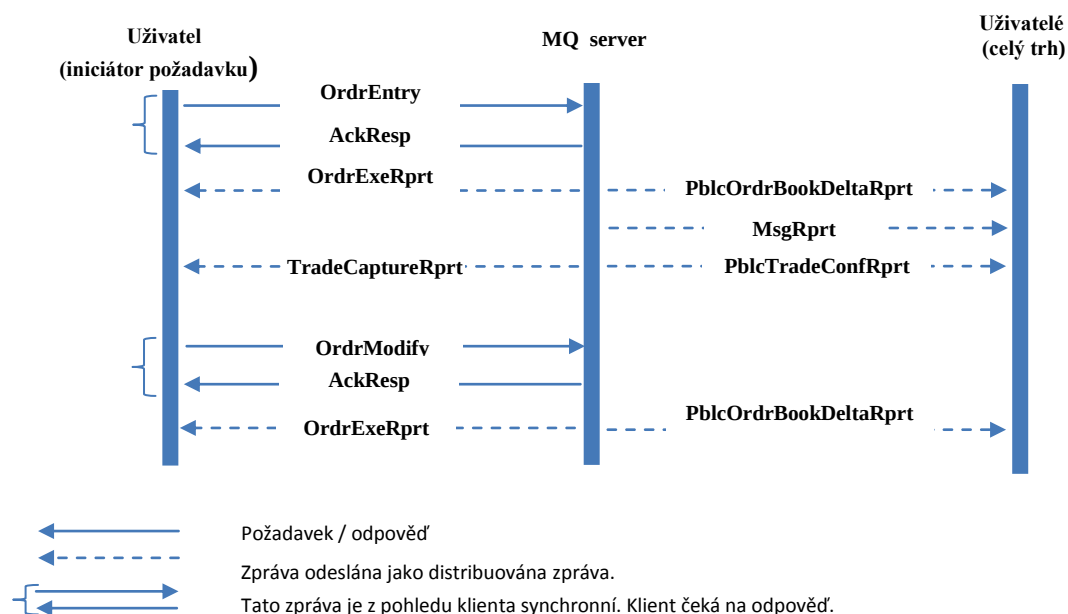
2.4.2. Práce s nabídkami

Uživatel zadá nabídku pomocí požadavku *OrdEntry* (případně modifikaci prostřednictvím *OrdModify*) a aplikační server odpoví pomocí *AckResp*, že požadavek úspěšně přijal nebo *ErrResp* v případě chybné definice zprávy. Po zpracování nabídky server odesílá klientovi zprávu o výsledku zavedení/modifikace nabídky prostřednictvím *OrdExeRprt*.

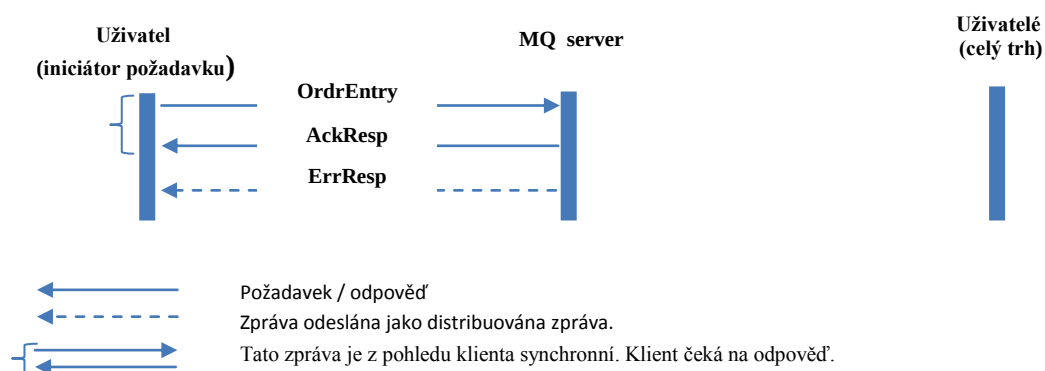
Následně je odeslána veřejná zpráva *PblcOrdBookResp* na všechny uživatele o změně vývěsky, pokud zavedení nabídky bylo úspěšné.

V případě, že dojde ke vzniku obchodu, je vlastníku nabídky odeslána zpráva *TradeCaptureRprt* a na všechny uživatele veřejná zpráva *MsgRprt* a *PblcTradeConfRprt*. Při vzniku obchodu odchází i na vlastníka protinabídky zprávy *OrdExeRprt* a *TradeCaptureRprt*.

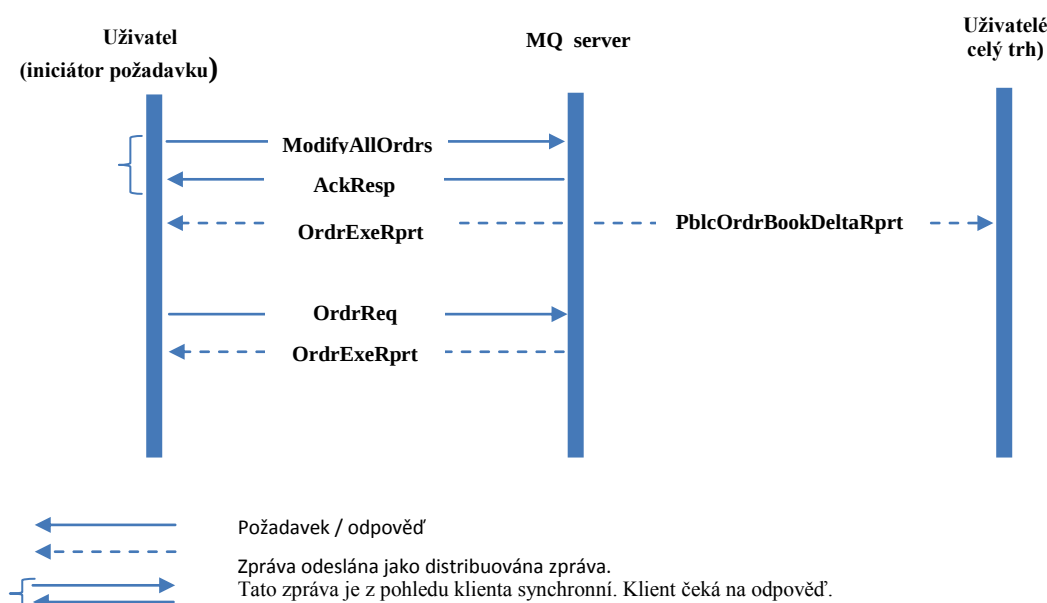
Je zde znázorněna i možnost dotazu na nabídky prostřednictvím *OrdReq*.



Obrázek 4 – Sekvenční schéma zavedení nabídky s jejím zobchodováním a modifikace nabídky bez vzniku obchodu



Obrázek 5 – Sekvenční schéma zadání neúspěšné nabídky

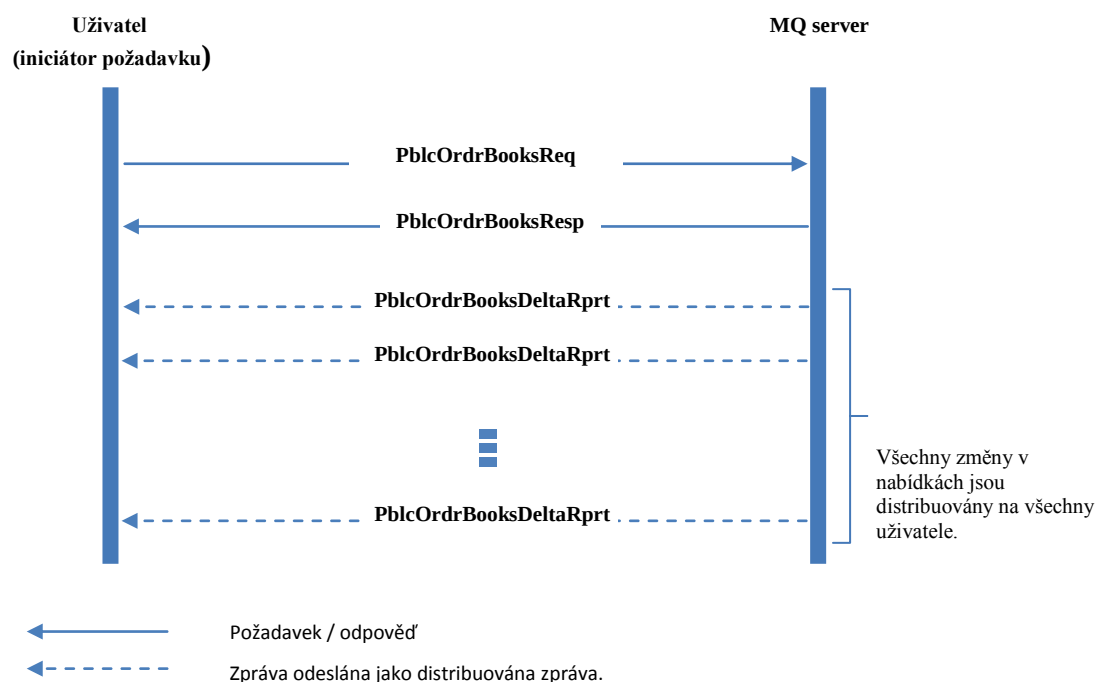


Obrázek 6 – Sekvenční schéma hromadné modifikace nabídek (deaktivace) a následný dotaz na nabídky

Picture 6 - Sequential scheme of mass bid modification (deactivation) and subsequent request for bids

2.4.3. Dotaz na veřejná data nabídek

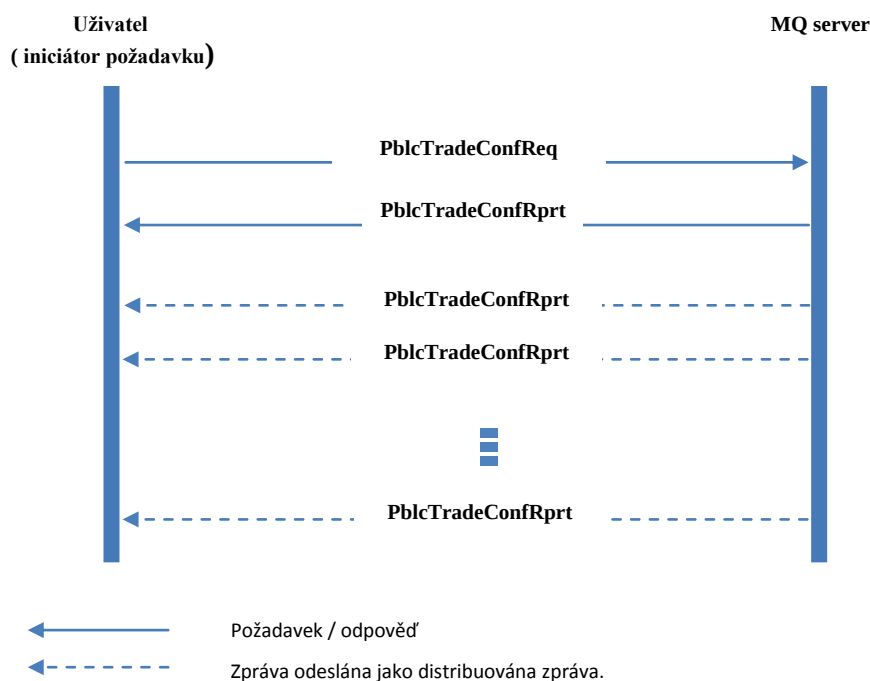
Uživatel zašle po přihlášení jednorázově požadavek na seznam aktivních nabídek na trhu prostřednictvím *PblcOrdrBooksReq* a server odpoví opisem nabídek *PblcOrdrBooksResp*. Tím klient obdrží plnou sadu aktivních nabídek v systému. Pokud došlo k zavedení nové nabídky nebo modifikaci, dojde k odeslání hromadné zprávy *PblcOrdrBooksDeltaRprt*.



Obrázek 7 – Sekvenční schéma provedení dotazu na nabídky

2.4.4. Dotaz na veřejná data obchodů

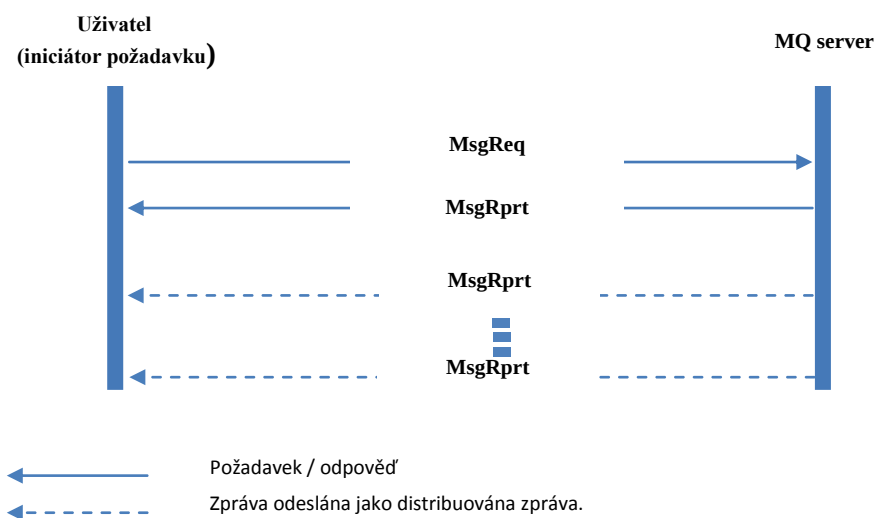
Uživatel zašle požadavek na obchody vzniklé na trhu prostřednictvím *PblcTradeConfReq* a server odpoví opisem obchodů *PblcTradeConfRprt*. Následují zprávy ze serveru v případě vzniku obchodu.



Obrázek 8 – Sekvenční schéma provedení dotazu na obchody

2.4.5. Dotaz na Informativní zprávy

Uživatel po úspěšném přihlášení si pošle dotaz na server s požadavkem *MsgReq* na seznam zpráv. V požadavku může specifikovat zda chce jen privátní zprávy nebo veřejné. Obdrží dané zprávy za požadovaný čas *MsgRprt* a dále již mu chodí distribuce nových zpráv automaticky.

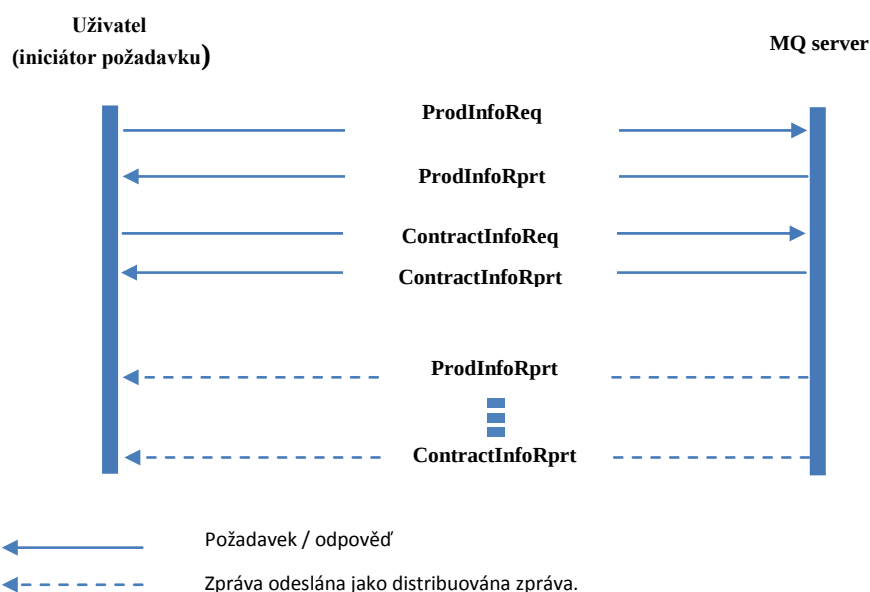


Obrázek 9 – Sekvenční schéma provedení dotazu na Zprávy trhu

2.4.6. Dotazy na Produkty a Kontrakty trhu

Uživatel si může vyžádat seznam platných produktů prostřednictvím dotazu *ProdInfoReq* a odpověď přijde zprávou *ProdInfoRprt*. V případě změny produktu, přijde aktualizace opět zprávou *ProdInfoRprt*.

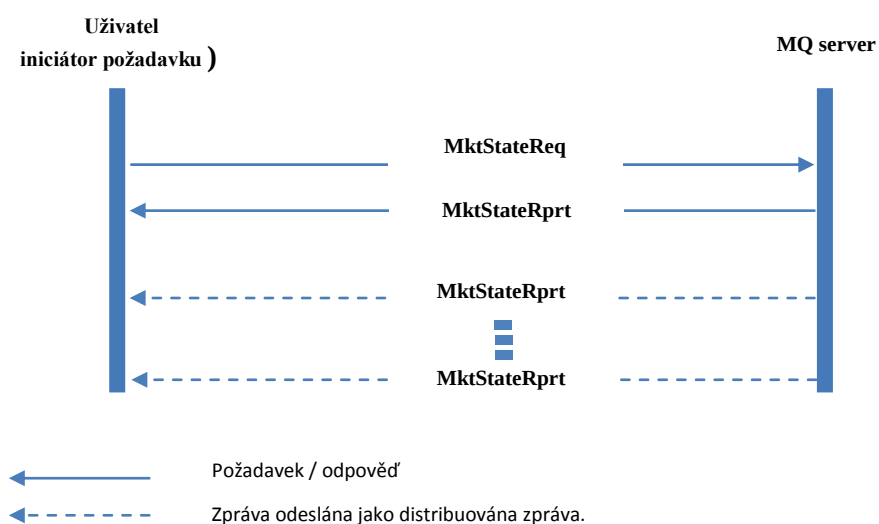
Obdobně jsou zprávy pro informace o Kontraktech. Uživatel si může vyžádat seznam platných kontraktů prostřednictvím dotazu *ContractInfoReq* a odpověď přijde zprávou *ContractInfoRprt*. V případě změny kontraktu, přijde aktualizace opět zprávou *ContractInfoRprt*.



Obrázek 10 – Sekvenční schéma provedení dotazu na Produkty a Kontrakty

2.4.7. Dotaz na Stav trhu

Uživatel si může vyžádat informace o aktuálním stavu trhu prostřednictvím dotazu *MktStateReq* a odpověď přijde zprávou *MktStateRprt*. V případě změny stavu trhu, přijde aktualizace opět zprávou *MktStateRprt*. Tyto zprávy slouží k zjištění aktuálního stavu trhu, zda není „Deaktivován“ – obchodování je pozastaveno.



Obrázek 11 – Sekvenční schéma provedení dotazu na Stav trhu

2.5. Komunikační zprávy

Všechny zprávy posílané mezi uživatelem a VDT/VT aplikací mají vlastní obsah zprávy definován pomocí XML formátu. Popis jednotlivých zpráv je uveden v následujících kapitolách.

2.5.1. Obecné informace

2.5.1.1. AMQP atributy

Atributy AMQP použité pro komunikaci mezi klientem a VDT/VT aplikací.

AMQP Message Atribut	Popis
content-type	Contains information about the used XML payload version as well as the used message type. Valid content-type definitions are (version number has to be filled with the used version): ▪ market/request; version=x (Used by the client when sending requests) ▪ market/response; version=x ▪ market/broadcast; version=x ▪ market/heartbeat; version=x ▪ market/error; version=x Current version of messages is 2.
reply-to	contains the queue name a response has to be sent to
user-id	contains the login-id of the logged in system
correlation-id	contains the request message id generated by client
expiration	contains an optional entry specifying if the request should be deleted if not executed within the specified time
contentEncoding	contains gzip, if messages are compressed (content is encrypted using gzip method); property is null if messages are not compressed. Message compressing can be activated per message type (e.g. OrdExeRprt) .
market-group-sequence	Identify the order of the broadcasts counted for „market-group-id“. Only for broadcast message.
market-group-id	Identification of routing key belongs to attribute „market-group-sequence“. Only for broadcast message.
timestamp	Timestamp of distributed message fulfilled by RabbitMQ server. For more information you can see at https://www.rabbitmq.com/releases/rabbitmq-java-client/v3.6.1/rabbitmq-java-client-javadoc-3.6.1/com/rabbitmq/client/AMQP.BasicProperties.html#getTimestamp() .

Tabulka 2 – Atributy zprávy dle AMQP

2.5.1.2. Konvence XML

V definici zpráv jsou použité následující konvence:

Tagy element jsou použity pro definování datových struktur. Data jsou obvykle uvedené v attributech. Typy: Elementy jsou zvýrazněné tučně, kdežto atributy nejsou zvýrazněné tučně **SE**: Structure Element. Data nejsou uváděné mezi tagy, ale mohou obsahovat atributy. (šedé pozadí, tučně) **CE**: Content Element. Data jsou vložena mezi tagy, mohou také obsahovat atributy (tučně). **A**: Atributy Elementů. Pořadí elementů a atributů není zaručeno a může se změnit.

2.5.1.3. Hodnoty množství ve zprávách

Hodnoty množství ve všech zprávách jsou uváděné jako celočíselné. Vlastní hodnota je dána skupinou atributů ve zprávě ProdInfoRprt - *decShftQty*, *smallestTrdUnit* a *qtyUnit* (viz. kapitola 2.5.4.13).

Atribut *decShftQty* určuje pozici desetinné čárky v zadaném celočíselném čísle (např. hodnota množství 5200 s atributem *decShftQty* = 3, znamená hodnotu 5,200).

Atribut *smallestTrdUnit* určuje nejmenší krok pro zadání množství (např. *smallestTrdUnit* = 100 a *decShftQty* = 3 znamená, že množství je možné zadávat s krokem 0,1).

Atribut *qtyUnit* definuje jednotku množství

2.5.1.4. Hodnoty ceny ve zprávách

Hodnoty týkající se cen jsou ve všech zprávách uváděné jako celočíselné. Vlastní hodnota je dána skupinou atributů ve zprávě ProdInfoRprt - *decShftPx*, *tickSize* a *currency* (viz. kapitola 2.5.4.13).

Atribut *decShftPx* určuje pozici desetinné čárky v zadaném celočíselném čísle (např. hodnota množství 3624 s atributem *decShftPx* = 2, znamená hodnotu 36,24).

Atribut *tickSize* určuje nejmenší krok pro zadání ceny (např. *tickSize* = 1 a *decShftPx* = 2 znamená, že cenu je možné zadávat s krokem 0,01).

Atribut *currency* definuje měnu pro obchodování.

2.5.1.5. Formát datumových položek ve zprávách

Datumové položky jsou definované jako “DateTime” typy. Formát těchto položek v XML zprávách je následující:

YYYY-MM-DDThh:mm:ssZ (2016-03-18T16:32:03Z)

Symbol	Popis	Příklad
YYYY	Rok	2016
MM	Měsíc	03
DD	Den	18
T	Oddělovač sekce datumu a času	T
hh	Hodina (0-23 h)	16
mm	Minuta	32
ss	Sekunda	03
Z	Nulová časová zóna = UTC čas	Z

Všechny datумы a časy jsou uváděné výhradně v UTC.

2.5.1.6. Heartbeat zpráva

Heartbeat zpráva obsahuje text s atributy “server-timestamp” a „interval-length“. Oba atributy jsou v milisekundách. První představuje rozdíl mezi aktuálním časem a datem 1.1.1970 0:00:00 UTC.

Příklad zprávy: server-timestamp=1468251175238;interval-length=30000

2.5.1.7. Standardní hlavička zprávy

Každá zpráva obsahuje standardní hlavičku s následujícími atributy.

XML Tag	Type	m/o	No.	Data Type	Short description
StandardHeader	SE	m		Structure	
marketID	A	m		Char(4)	Market Identification Code (MIC) of the market to which the request is sent or from which the request originates. The following values are allowed: "IM": Intraday market. "BALM": Balancing market
clientData	SE	o	0..1	Structure	
clientDataInt	A	o		Integer	The client data fields in this section can be used by the client to store information or meta-data about a request.
clientDataString	A	o		String	
clientCorrelationId	A	o		String	
					The content in these fields is not used by CS OTE system. Content is send back to client in response.

Tabulka 3 – Hlavička zprávy

2.5.1.8. Popis parametrů jednotlivých zpráv

V následujících kapitolách jsou definované následující parametry zpráv:

- Typ - typ zprávy
 - Inquiry Request – dotaz na data
 - Management request – výkonný pokyn
 - Broadcast – hromadná zpráva
- Role – přístupnost zprávy dle role
- Směrovací klíč (Routing key) – směrování zprávy na MQ serverLimit zpráv – max. počet zpráv daného názvu za definovaný čas, které budou serverem zpracovány pro jednotlivé uživatele, aniž by byly odmítnuté. Definice formátů a/b, přičemž „a“ představuje max. počet zpráv za 1 minutu a „b“ max. počet zpráv za 1 hodinu. Pokud není limit uveden, počet zpráv není omezen. Limit je počítán odděleně pro každé marketID.

2.5.2. Obecné dotazy a odpovědi

2.5.2.1. Login Request (LoginReq)

LoginReq	
Type:	Inquiry Request
Roles:	<All>
Routing Keys:	market.request.inquiry
Request Limits:	3/20

Požadavek na přihlášení do systému . Systém odpoví zprávou „UserReport“.

XML Tag	Type	m/o	No.	Data Type	Short description
LoginReq	SE			Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
user	A	m		String	Login ID of the user that want to login to the CS OTE system.
force	A	m		Boolean	Flag that indicates if this user want to force a login even if a user with the same credentials is already logged in into the CS OTE system.
disconnectAction	A	m		String	Action that will be executed in case of an unexpected connection loss

					between user and CS OTE system, irrespective of where the connection loss will be (user – AMQP – CS OTE system). The following values are allowed: "NO": No action is executed. "DEACT_USER_ORDRS": All orders of this user will be deactivated.
--	--	--	--	--	---

Tabulka 4 – Struktura zprávy Login Request

2.5.2.2. User Report (UserRprt)

UserRprt	
Type:	Management Response, Broadcast
Response to:	LoginReq (sent to the user-generated private response queue or a broadcast to market.broadcastQueue.<login-id>)
Broadcasted:	Yes
Broadcast Routing Keys:	USR_<login-id>
Roles:	<All>

Zpráva obsahuje základní atributy uživatele. „User Report“ je vrácen jako odpověď na „Login Request“ a je také distribuován při změně konfigurace přiřazení uživatele k produktům.

XML Tag	Type	m/o	No.	Data Type	Short description
UserRprt	SE	m		Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapte 2.5.1.7.
usrId	A	m		Integer	The unique identifier of a user.
sessionId	A	m		Long	The current session id of the user given after login to the system.
state	A	m		Char(4)	Current state of the User. The following values are allowed: "ACTI": User is active. It is possible to trade using this User. "DELE": User is deleted. Trading using this User is not possible. "SUSP": User is suspended. Trading using this User is not possible.
prtId	A	m		Integer	The participant id the user belongs to.
name	A	m		String	Name of the user.
connectionLossMsg	A	o		String	In case of a connection loss for the previous user session, this field is filled with a connection loss message, indicating the connection loss event with date and time and the logout action executed by the CS OTE system.
Assgs	SE	m	1	Structure	
usrRole	CE	o	0..n	String	Contains the user roles assigned to the user
prdAssg	CE	o	0..n	String	Contains the products for the user
DlvryArea	SE	o	0..n	Structure	Delivery Area
dlvryAreaId	A	m		String	Delivery Area Id.
revisionNo	A	m		Long	Revision number. With every change of the delivery area this value is increased by one.
state	A	m		Char(4)	Current state of the delivery area. The following values are allowed: "IACT": Delivery area is inactive and thus not tradable. "ACTI": Delivery area is active. It is possible to trade in that area.
name	A	m		String	Name of the delivery area usually used for display purposes.
longName	A	m		String	Long name of the delivery area usually.
prodName	CE	o	0..n	String	List of assigned products to the delivery area.

Tabulka 5 – Struktura zprávy User Report

2.5.2.3. Logout Request (LogoutReq)

LogoutReq	
Type:	Inquiry Request
Roles:	<All>
Routing Keys:	market.request.inquiry
Request Limits:	3/20

Požadavek na odhlášení uživatele ze systému.

XML Tag	Type	m/o	No.	Data Type	Short description
LogoutReq	SE			Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
sessionId	A	m		Long	Session id of the PX session passed to the PX on login.

Tabulka 6 – Struktura zprávy Logout Request

2.5.2.4. Logout Report (LogoutRprt)

LogoutRprt					
Type:	Inquiry Response, Broadcast				
Response to:	LogoutReq (sent to the user-generated private response queue or a broadcast to market . broadcastQueue.<login-id>)				
Broadcast:	Yes				
Broadcast Routing Keys:	USR_<login-id>				
Roles:	<All>				

Zpráva o odhlášení uživatele ze systému. Je odeslána jako odpověď na požadavek o odhlášení "Logout Request" nebo hromadná zpráva jako důsledek konkurenčního přihlášení stejného uživatele s vynuceným přihlášením (force=true)

XML Tag	Type	m/o	No.	Data Type	Short description
LogoutRprt	SE			Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
sessionId	A	m		Long	Session id of the PX session passed to the PX on login.
usrId	A	m		Integer	User ID identification.
txt	A	o		String	Text field containing information about the reason of the logout.

Tabulka 7 – Struktura zprávy Logout Report

2.5.2.5. Acknowledgement Response (AckResp)

AckResp					
Type:	Management Response				
Response to:	OrdEntry; OrderModify; ModifyAllOrders: (sent to the user-generated private response queue)				
Broadcast:	No				
Routing Keys:	---				
Roles:	<All>				

Potvrzující zpráva o přijetí pokynu ke zpracování.

XML Tag	Type	m/o	No.	Data Type	Short description
AckResp	SE			Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.

Tabulka 8 – Struktura zprávy Acknowledgement Report

2.5.2.6. Error Response (ErrResp)

ErrResp					
Type:	Inquiry Response; Management Response; Broadcast				
Response to:	<All> (sent to the user-generated private response queue or a broadcast to market .				

	broadcastQueue.<login-id>)
Broadcast:	Yes
Broadcast Routing Keys:	USR_<login-id>
Roles:	<All>

Chybová zpráva distribuována v případě neúspěšného provedení pokynu/dotazu.

XML Tag	Type	m/o	No.	Data Type	Short description
ErrResp	SE			Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
Error	SE	m	1..n	Structure	
errCode	A	m		Integer	Predefined error codes. Some error messages do not have a specific error code. In this case the value is 0.
errEn	A	m		String	The error message for this error – English version.
errCz	A	m		String	The error message for this error – Czech version.
clOrdId	A	o		Char(40)	Client order ID.

Tabulka 9 – Struktura zprávy Error Report

2.5.3. Zavedení a správa nabídek

2.5.3.1. Order Entry (OrdEntry)

OrdEntry	
Type:	Management Request
Roles:	EmtasImIns, EmtasBalmIns
Routing Keys:	market.request.management

Zavedení jedné nebo více nabídek. Max. počet nabídek v rámci jedné zprávy je 25.

XML Tag	Type	m/o	No.	Data Type	Short description
OrdEntry	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
OrdList	SE	m	1	Structure	List of all orders contained in the basket.
Ord	SE	m	1..25	Structure	
state	A	o		Char(4)	“ ACTI ”: The order is entered and immediately exposed to the market for execution. This is the default value. “ HIBE ”: The order is entered into the CS OTE system but not exposed to the market.
validityRes	A	o		Char(3)	Validity restriction of the order. If this field is omitted, the order will be treated as a “Good for Session” order. Valid values: “ GFS ” (Good for trading session): The order rests in the order book until it is either executed, removed by the user or the current trading session (trading phase) of the underlying contract ends. “ GTD ”: The order rests in the order book until the date specified in the validityDate field. “ NON ” (No validity restriction): Mandatory for orders with the execution restriction “FOK” or “IOC”.

XML Tag	Type	m/o	No.	Data Type	Short description
validityDate	A	o		DateTime	This field is mandatory in case of validityRes equals “GTD”. It is used to define the date until which the order is valid. The remaining part of the order will be removed from the order book after this point in time.
txt	A	o		String	Comment entered by the user. Maximum possible length is 250 characters.
type	A	m		Char(1)	Order type. Valid values: “O”: Regular limit order (for all predefined contracts). “I”: Iceberg order. “B”: User defined block order.
dlvryAreaId	A	m		String	Defines the delivery area of the order. Valid value is “CZ”.
ordrExeRestriction	A	o		Char(3)	Execution restriction of the order. Valid values: “NON”: No restriction. This is the default. “FOK” (Fill or Kill): The order is immediately fully executed or deleted. “IOC” (Immediate and cancel): The order is executed immediately to its maximum extend. In case of a partial execution, the remaining volume is removed from the order book. “AON” (All or None): The order must be filled completely or not at all. The order stays in the order book until it is executed or removed by the system or user. This execution restriction can be used only in combination with User Defined Block Orders (for which only AON execution restriction is allowed).
qty	A	m		Integer	Contains the total quantity of the order. In case of an Iceberg order this field corresponds to the hidden quantity + display quantity.
displayQty	A	o		Integer	Used to define display quantity of an Iceberg Order. This field is required only in the case of type=’I’.
px	A	o		Long	Limit price of the order in currency defined by contracts. Value is multiplied by 100, e.g. 1 Euro = 100.
ppd	A	o		Long	Peak price delta for Iceberg orders. - The ppd of buy orders must be smaller or equal than zero. - The ppd of sell orders must be greater or equal than zero. If it is omitted the system will assume a value of “0,00”.
side	A	m		String	Defines on which side of the market the order is entered (“BUY”, “SELL”).
contract	A	o		String	Contract code identifier. Applicable for orders for pre-defined contracts only.
dlvryStart	A	o		DateTime	Start of delivery of the underlying contract. Applicable for User Defined Block Orders only. The attribute is ignored if <i>contract</i> is filled in.
dlvryEnd	A	o		DateTime	End of delivery of the underlying contract. Applicable for User Defined Block Orders only. The attribute is ignored if <i>contract</i> is filled in.
clOrdId	A	o		String	Client Order Id with a maximum length of 40 characters.

Tabulka 10 – Struktura zprávy Order Entry Message

2.5.3.2. Order Modify (OrdrModify)

OrdrModify	
Type:	Management Request
Roles:	EmtasImIns, EmtasBalmIns
Routing Keys:	market.request.management

Zpráva pro modifikaci jedné nebo více nabídek. Max. počet nabídek v rámci jedné zprávy je 25.

XML Tag	Type	m/o	No.	Data Type	Short description
OrdrModify	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
ordrModType	A	m		Char(5)	Offers the possibility to activate, deactivate, modify or delete all orders contained in the basket. “ACTI” : Activate all orders contained in this basket. Already active orders are ignored. “HIBE” : Deactivates (hibernates) all orders contained in the basket. Hibernated orders are removed from the order book but are still available for modification or activation in the own orders list. “MODI” : Modifies all orders in the basket. “DELE” : Deletes all orders in the basket.
OrdrList	SE	m	1	Structure	List of all orders contained in the basket.
Ordr	SE	m	1..25	Structure	Definition of a single order.
validityRes	A	o		Char(3)	Validity restriction of the order. If this field is omitted, the order will be treated as a “Good for Session” order. Valid values: “GFS” (Good for trading session): The order rests in the order book until it is either executed, removed by the user or the current trading session (trading phase) of the underlying contract ends. “GTD” (Good till date): The order rests in the order book until the date specified in the validityDate field. “NON” (No validity restriction): Mandatory for orders with the execution restriction “FOK” or “IOC”.
validityDate	A	o		DateTime	This field is mandatory in case of validityRes equals “GTD”. It is used to define the date until which the order is valid. The remaining part of the order will be removed from the order book after this point in time.
type	A	m		Char(1)	Order type. It must be the same as order type of original order. Order type can’t be changed by modification. Valid values: “O” : Regular limit order (for all predefined contracts). “I” : Iceberg order. “B” : User defined block order.
txt	A	o		String	Comment entered by the user. Maximum possible length is 250 characters.
ordrExeRestriction	A	o		Char(3)	Execution restriction of the order. Valid values: “FOK” (Fill or Kill): The order is immediately fully executed or deleted. “IOC” (Immediate and cancel): The order is executed immediately to its maximum extend. In case of a partial execution, the remaining volume is removed from the order book. “NON” : No restriction. This is the default. “AON” (All or None): The order must be filled completely or not at all. The order stays in the order book until it is executed or removed by the system or user. AON execution restriction can be used only in combination with User Defined Block Orders (for which only AON execution restriction is allowed) and hence can’t be changed by modification.
qty	A	m		Integer	Contains the total quantity of the order. In case of an Iceberg order this field corresponds to the hidden quantity + display quantity.
displayQty	A	o		Integer	Used to define display quantity of an Iceberg Order.
px	A	o		Long	Limit price of the order in currency defined by contract. Value is multiplied by 100, e.g. 1 Euro = 100.
ppd	A	o		Long	Peak price delta for Iceberg orders. - The ppd of buy orders must be smaller or equal than zero. - The ppd of sell orders must be greater or equal than zero. If it is omitted the system will assume a value of “0,00”.
ordrId	A	m		Long	Order Id as returned by the CS OTE system. This value is used to identify the order to be modified.
revisionNo	A	m		Long	The latest revision number of the order must be provided by the user. In case the CS OTE has another revision number of currently valid order, it will reject the request with an ErrResp.
clOrdrId	A	o		String	Client Order Id with a maximum length of 40 characters.

Tabulka 11 – Struktura zprávy Order Modify Message

2.5.3.3. Order Request (OrdReq)

OrdReq	
Type:	Inquiry Request
Roles:	EmtasImTsAcc, EmtasBalmTsAcc
Routing Keys:	market.request.inquiry
Request Limits:	1/10

Dotaz na stav vlastních nabídek.

XML Tag	Type	m/o	No.	Data Type	Short description
OrdReq	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
contract	CE	o	0..1000	String	List of contract codes If no contract code is given, the own orders for all contracts assigned to the requesting user are returned.

Tabulka 12 – Struktura zprávy Order Request

2.5.3.4. Order Execution Report (OrdExeRprt)

OrdExeRprt	
Type:	Management Response; Broadcast
Response to:	OrdEntry; OrdModify; OrdReq; ModifyAllOrds; (sent to the user-generated private response queue or a broadcast to market.broadcastQueue.<login-id>)
Broadcast:	Yes
Broadcast Routing Keys:	<prodName>.<particId>
Roles:	EmtasImTsAcc, EmtasBalmTsAcc

Zpráva o úspěšné modifikaci nabídky. Zpráva je odeslána účastníkům trhu v následujících případech:

- úspěšné zavedení nabídky
- úspěšná modifikace nabídky
- částečné nebo úplné zobchodování nabídky
- jako odpověď dotazu na nabídku (pouze v tomto případě je odeslána do privátní fronty pro odpovědi, v ostatních případech je odeslána do fronty pro hromadné zprávy)

XML Tag	Type	m/o	No.	Data Type	Short description
OrdExeRprt	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
OrdList	SE	o	0..1	Structure	
Ord	SE	o	0..n	Structure	
action	A	m		String	Code of the last action provided on the order. Valid values are: “UADD”: Order added by user. “UHIB”: Order hibernated by user. “UMOD”: Order modified by user. “UDEL”: Order deleted by user. “SHIB”: Order hibernated by the system. “SMOD”: Order modified by the system. “SDEL”: Order deleted by the system. “FEXE”: Order is fully executed. If an order comes into the system and gets executed immediately by matching an already existing order only one OrdExeRprt for this order is sent with action FEXE or PEXE. If an order comes into the system and gets executed by a later entered order two messages are sent. One for the order entry with UADD and later one for the execution with either FEXE or PEXE. “PEXE”: Partial execution of order. “IADD”: A new slice of an Iceberg order was added to the service.
validityRes	A	o		Char(4)	Validity restriction of the order. If this field is omitted, the order will be treated

XML Tag	Type	m/o	No.	Data Type	Short description
					as a “Good for Session” order. Valid values: “GFS” (Good for trading session): The order rests in the order book until it is either executed, removed by the user or the current trading session (trading phase) of the underlying contract ends. “GTD” (Good till date): The order rests in the order book until the date specified in the vldtyDate field. “NON” (No validity restriction): Mandatory for orders with the execution restriction “FOK” or “IOC”.
validityDate	A	o		DateTime	This field is mandatory in case of validityRes equals “GTD”. It is used to define the date until which the order is valid. The remaining part of the order will be removed from the order book after this point in time.
timestmp	A	m		DateTime	Timestamp of the order entry as determined by the CS OTE system. This timestamp determines the execution priority in case of identical limit prices.
revisionNo	A	m		Long	This value is increased in case of a partial execution, hibernation, modification without execution priority change.
usrCode	A	m		String	User code of the user who entered the order.
state	A	m		Char(4)	The current state of the order in the system. Valid values: “HIBE” : The order is entered into the XBID SOB system but not exposed to the market. “ACTI” : The order is entered and immediately exposed to the market for execution “IACT” : The order is inactive due time validity or fully executed. “DELE” : The order is deleted
type	A	m		Char(1)	Order type. Valid values: “O” : Regular limit order (for all predefined contracts). “I” : Iceberg order. “B” : User defined block order.
dlrvyAreaId	A	m		String	Defines the delivery area of the order. Valid value is “CZ”.
txt	A	o		String	Comment entered by the user. Maximum possible length is 250 characters.
ordrExeRestriction	A	o		Char(3)	Execution restriction of the order. Valid values: “FOK” (Fill or Kill): The order is immediately fully executed or deleted. “IOC” (Immediate and cancel): The order is executed immediately to its maximum extend. In case of a partial execution, the remaining volume is removed from the order book. “NON” : No restriction. “AON” (All or None): The order must be filled completely or not at all. The order stays in the order book until it is executed or removed by the system or user. AON execution restriction can be used only in combination with User Defined Block Orders (for which only AON execution restriction is allowed) and hence can't be changed by modification.
totalQty	A	m		Integer	The total quantity entered with this order. If the order is partially matched, the totalQty still contains the original quantity value.
qty	A	m		Integer	Contains the quantity exposed to the market. In case of an Iceberg Order this is the rest of the display quantity.
hiddenQty	A	o		Integer	Contains the hidden quantity of the Iceberg order. The total executable quantity may be calculated by adding the hiddenQty to the qty.
displayQty	A	o		Integer	Used to define display quantity of an Iceberg Order.
px	A	o		Long	Limit price of the order in currency defined by contract. Value is multiplied by 100, e.g. 1 Euro = 100.
ppd	A	o		Long	Peak price delta for Iceberg orders.
side	A	m		String	Defines on which side of the market the order is entered. Valid values: “BUY” : Buy order. “SELL” : Sell order.
contract	A	m		String	Contract code identifier.
ordrId	A	m		Long	Order Id as returned by the CS OTE system.
lastUpdateUsrCode	A	m		String	Information about the user who last updated the order
clOrdrId	A	o		String	Client Order Id with a maximum length of 40 characters. This value is not modified by the CS OTE system and may be used by Client applications to identify orders.

Tabulka 13 – Struktura zprávy Order Execution Report

2.5.3.5. Modify All Orders (ModifyAllOrdrs)

ModifyAllOrdrs	
Type:	Management Request
Roles:	EmtasImIns, EmtasBalmIns
Routing Keys:	market.request.management

Zpráva pro hromadnou aktivaci, deaktivaci a zrušení nabídek.

XML Tag	Type	m/o	No.	Data Type	Short description
ModifyAllOrders	SE			Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
prtclId	A	o		String	Unique identifier of a partic.
usrId	A	o		Integer	Unique identifier of a user.
ordrModType	A	m		Char(4)	Modification type for the orders: “ACTI” : Activate all orders. Already active orders are ignored. “HIBE” : Deactivates (hibernates) all orders. Hibernated orders are removed from the order book but are still available for modification or activation in the own orders list. “DELE” : Deletes all orders.
contract	CE	o	0..1000	String	List of contract codes. If no contract code is given, the own orders for all contracts assigned to the specified participant or user are changed.

Tabulka 14 – Struktura zprávy Modify All Orders Message

2.5.4. Informace o trhu

2.5.4.1. Public Order Books Request (PbclOrdrBooksReq)

PbclOrdrBooksReq	
Type:	Inquiry Request
Roles:	<All>
Routing Keys:	market.request.inquiry
Request Limits:	2/20

Požadavek na vývěsku požadovaného kontraktu.

XML Tag	Type	m/o	No.	Data Type	Short description
PbclOrdrBooksReq	SE		1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
contractType	A	(m)		Char(3)	Defines which kind of contracts should be retrieved: Possible values are: “ALL” – All kind of contracts (pre-defined and user-defined) “PDC” – Only pre-defined contracts “UDC” – Only user-defined contracts This attribute is ignored when contractId is specified.
prodName	CE	(m)	0..1000	String	List of product names. All order books for these products are returned. Delivery area may be specified to filter the result. Please note: If no product name is given, at least one contract (see below) must be provided.
contract	CE	(m)	0..1000	String	List of contract codes. Please note: If no contract is given, at least one product name (see above) must be provided. If both values are given the contract is taken.
dlvryAreaId	CE	o	0..1000	String	Delivery areas for which the order book(s) should be retrieved.

Tabulka 15 – Struktura zprávy Public Order Books Request

2.5.4.2. Public Order Books Response (PbclOrdrBooksResp)

PbclOrdrBooksResp	
Type:	Inquiry Response
Response to:	PbclOrdrBooksReq (sent to the user-generated private response queue)

Broadcast:	No
Broadcast Routing Keys:	---
Roles:	<All>

Veřejné informace o aktuálních nabídkách daného kontraktu.

Zpráva je distribuována jako odpověď na dotaz “Public Order Book Request”.

XML Tag	Type	m/o	No.	Data Type	Short description
PblcOrdrBooksResp	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
OrdrbookList	SE	o	0..1		
OrdrBook	SE	o	0..n	Structure	
revisionNo	A	m		Long	This value is increased in case of any change in the order book. Please note: revision numbers of order book are stored in memory only (not persistent) on CS OTE system. After a restart of CS OTE system, the revision numbers of order books will start from 0 again.
contract	A	m		String	Contract code identifier.
dlvryAreaId	A	m		String	Delivery Area to which the attached order books refer to.
lastPx	A	o		Long	Last traded price.
pxDir	A	o		Integer	Defines the direction of the price movement with regard to the last 2 trades happened and that are relevant for this orderbook. Valid values are: -1: Price decreased 0: Price unchanged 1: Price increased
lastQty	A	o		Integer	Last traded quantity.
totalQty	A	o		Long	The total quantity traded during this trading session.
lastTradeTime	A	o		DateTime	Timestamp of the last execution.
highPx	A	o		Long	Highest traded price since the start of the trading period.
lowPx	A	o		Long	Lowest traded price since the start of the trading period.
SellOrdrList	SE	o	0..1	Structure	
OrdrBookEntry	SE	o	0..n	Structure	
ordrId	A	m		Long	Order Id as determined by the CS OTE system.
qty	A	m		Integer	The quantity of the order which is exposed in that delivery area.
px	A	m		Long	Limit price of the order in currency defined by contract. Value is multiplied by 100, e.g. 1 Euro = 100.
ordrEntryTime	A	m		DateTime	Timestamp of the order.
ordrExeRestriction	A	o		Char(3)	Execution restriction of the order. This attribute is set only in case of AON orders (value = “AON”).
ordrType	A	o		Char(1)	“O”: Regular limit order. “I”: Iceberg order. “B”: User defined block order.
BuyOrdrList	SE	o	0..1	Structure	
OrdrBookEntry	SE	o	0..n	Structure	
ordrId	A	m		Long	Order Id as determined by the CS OTE system.
qty	A	m		Integer	The quantity of the order which is exposed in that delivery area.
px	A	m		Long	Limit price of the order in currency defined by contract. Value is multiplied by 100, e.g. 1 Euro = 100.
ordrEntryTime	A	m		DateTime	Timestamp of the order.
ordrExeRestriction	A	o		Char(3)	Execution restriction of the order. This attribute is set only in case of AON orders (value = “AON”).
ordrType	A	o		Char(1)	“O”: Regular limit order. “I”: Iceberg order. “B”: User defined block order.

Tabulka 16 – Struktura zprávy Public Order Books Report

2.5.4.3. Public Order Books Delta Report (PblcOrdrBooksDeltaRprt)

PblcOrdrBooksDeltaRprt	
Type:	Broadcast
Response to:	n/a
Broadcast:	Yes

Broadcast Routing Keys:	<prodName>
Roles:	<All>

Zpráva Public Order Book Delta Report je zaslána při zavedení nebo změně aktivní nabídky. Zpráva obsahuje všechny změněné nabídky od předchozí distribuce zprávy *PblcOrdrBooksDeltaRprt* pro daný kontrakt.

Formát zprávy je shodný se zprávou *PblcOrdrBooksResp*.

2.5.4.4. Message Request (MsgReq)

MsgReq	
Type:	Inquiry Request
Roles:	<ALL>
Routing Keys:	market.request.inquiry
Request Limits:	1/10

Dotaz na zprávy obchodního systému, které vznikly na obchodním systému v minulosti. Možné se dotazovat maximálně na zprávy za 1 den zpětně.

XML Tag	Type	m/o	No.	Data Type	Short description
MsgReq	SE		1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
type	A	m		Char(7)	Defines what kinds of messages are returned, allowing filtering the messages on a request level. Valid Values: "ALL": Return all messages. "PUBLIC": Return only public messages. "PRIVATE": Return only private messages.
endDate	A	m		DateTime	Timestamp defining to which point in time the messages should be retrieved.
startDate	A	m		DateTime	Timestamp defining from which point in time the messages should be retrieved. It is possible only to retrieve messages from the last 1 day.

Tabulka 17 – Struktura zprávy Message Request

2.5.4.5. Message Report (MsgRprt)

MsgRprt	
Type:	Inquiry Response, Broadcast
Response to:	MsgReq (sent to the user-generated private response queue or a broadcast to market.broadcastQueue.<login-id>)
Broadcasted:	Yes
Broadcast Routing Keys:	PRTC <particId> <prodName> <prodName>.PRTC_<particId> public
Roles:	<All>

Zprávy z obchodního systému jsou zaslány jako odpověď dotazu na zprávy “Message Request” a dále distribuovány při vzniku nové zprávy v obchodním systému.

XML Tag	Type	m/o	No.	Data Type	Short description
MsgRprt	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
MsgList	SE	o	0..1		
Msg	SE	o	0..n		
msgId	A	m		Integer	The message Id as assigned by the CS OTE system.

type	A	m			Defines the message type. Valid Values: "PUBLIC": The message is a public message. "PRIVATE": The message is a private message.
contract	A	o		String	Underlying contract.
timestmp	A	m		DateTime	Timestamp of the message as assigned by the CS OTE system.
svrty	A	m		String	Severity of the message: "URG": Urgent message. "ERR": Error. "HIG": High prioritized message. "MED": Medium prioritized message. "LOW": Low priority message.
mrktSupervisionMsg	A	m		Boolean	Determines if the message has been send by market supervision
txtEn	A	m		String	Message text. – English version.
txtCz	A	m		String	Message text – Czech version.
sellDlvryAreaId	A	o		String	In case of an order execution, this field contains the delivery area of the sell side.
buyDlvryAreaId	A	o		String	In case of an order execution, this field contains the delivery area of the buy side.

Tabulka 18 – Struktura zprávy Message Report

2.5.4.6. Trade Capture Request (TradeCaptureReq)

TradeCaptureReq	
Type:	Inquiry Request
Roles:	EmtasImTsAcc, EmtasBalmTsAcc
Routing Keys:	market.request.inquiry
Request Limits:	7/35

Dotaz na vlastní obchody. Možné se dotazovat max. 7 dní zpětně s maximálním rozpětím datumů 24 hodin (hodnoty omezení jsou systémově nastavitelné). V případě chybných vstupních parametrů je vrácena odpověď „ErrResp“.

XML Tag	Type	m/o	No.	Data Type	Short description
TradeCaptureReq	SE				
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
startDate	A	m		DateTime	Start of the period for which the trades are retrieved. This value must fulfil the following conditions: endDate – startDate <= 24 hours
endDate	A	o		DateTime	End of the period for which the trades are retrieved. The following condition must be fulfilled: endDate – startDate <= 24 hours If no end date is given, the CS OTE system will return all trades until midnight of the start date. In case of invalid value Error Message is returned stating that diff is bigger than max value.

Tabulka 19 – Struktura zprávy Trade Capture Request

2.5.4.7. Trade Capture Report (TradeCaptureRprt)

TradeCaptureRprt	
Type:	Inquiry Response, Broadcast
Response to:	TradeCaptureReq (sent to the user-generated private response queue or a broadcast to market. broadcastQueue.<login-id>)
Broadcasted:	Yes
Broadcast Routing Keys:	halftrade.<prodName>.PRTC_<particId>
Roles:	EmtasImTsAcc, EmtasBalmTsAcc

Zpráva o vzniku obchodu je odeslána na oba účastníky daného obchodu, přičemž pro každého je vyplněna jen ta část obchodu, která se ho týká. Zpráva je také odeslána jako odpověď na „Trade Capture Request“.

XML Tag	Type	m/o	No.	Data Type	Short description
TradeCaptureRprt	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
TradeList	SE	o	0..1		
Trade	SE	o	0..n	Structure	
tradeId	A	m		Long	Trade ID of the trade.
state	A	m		Char(4)	Current state of the trade. Valid values are: "CNCL": Trade was cancelled by Central Admin. "RREJ": Requested Recall was rejected by Central Admin "RGRA": Requested Recall was granted by Central Admin. "RREQ": Recall of this trade was requested. "ACTI": Trade is active (this is the default value).
contract	A	m		String	Contract code
qty	A	m		Integer	Executed quantity.
px	A	m		Long	Execution price in currency defined by contract. Value is multiplied by 100, e.g. 1 Euro = 100.
execTime	A	m		DateTime	Execution date as assigned by the CS OTE system.
Buy	SE	o	0..1	Structure	
ordrId	A	m		Long	Order Id of the buy side order.
dlvryAreaId	A	m		String	Delivery Area to which the attached order books refer to.
prtcId	A	m		String	Participant who entered the buy side order.
usrCode	A	m		String	User code of the user who entered the buy side order.
clOrdrId	A	o		String	Client's identification of order.
txt	A	o		String	Text of the buy side order.
Sell	SE	o	0..1	Structure	
ordrId	A	m		Long	Order Id of the sell side order.
dlvryAreaId	A	m		String	Delivery Area to which the attached order books refer to.
prtcId	A	m		String	Participant who entered the sell side order.
usrCode	A	m		String	User code of the user who entered the sell side order.
clOrdrId	A	o		String	Client's identification of order.
txt	A	o		String	Text of the sell side order.

Tabulka 20 – Struktura zprávy Trade Capture Report

2.5.4.8. Public Trade Confirmation Request (PblcTradeConfReq)

PblcTradeConfReq	
Type:	Inquiry Request
Roles:	EmtasImTsAcc, EmtasBalmTsAcc
Routing Keys:	market.request.inquiry
Request Limits:	7/35

Dotaz na veřejné informace o vzniklých obchodech. Možné se dotazovat max. 7 dní zpětně s maximálním rozpětím datumů 24 hodin (hodnoty omezení jsou systémově nastavitelné). V případě chybných vstupních parametrů je vrácena odpověď „ErrResp“.

XML Tag	Type	m/o	No.	Data Type	Short description
PblcTradeConfReq	SE	m		Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
startDate	A	m		DateTime	Start of the period for which the trades are retrieved. This value must fulfil the following conditions: endDate – startDate <= 24 hours
endDate	A	o		DateTime	End of the period for which the trades are retrieved. The following condition must be fulfilled: endDate – startDate <= 24 hours If no end date is given, the system will return all trades until midnight of the start date. In case of invalid value Error Message is returned stating that diff is bigger than max value.
prodName	CE	o	0..1000	String	Products for which the public trade confirmations are requested. If not supplied all products for which the user has access rights are returned

Tabulka 21 – Struktura zprávy Public Trade Confirmation Request

2.5.4.9. Public Trade Confirmation Report (PblcTradeConfRprt)

PblcTradeConfRprt	
Type:	Inquiry Response, Broadcast
Response to:	PblcTradeConfReq (sent to the user-generated private response queue or a broadcast to market . broadcastQueue.<login-id>)
Broadcasted:	Yes
Broadcast Routing Keys:	public.trade.<prodName>
Roles:	EmtasImTsAcc, EmtasBalmTsAcc

Zpráva o vzniku obchodu. Zpráva je distribuována na všechny uživatele, kteří mají přiřazený kontrakt, na kterém obchod vznikl. Zpráva je také odeslána jako odpověď na „Public Trade Confirmation Request“.

XML Tag	Type	m/o	No.	Data Type	Short description
PblcTradeConfRprt	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
TradeList	SE	m	1	Structure	
PblcTradeConf	SE	o	0..n	Structure	
tradeId	A	m		Long	Trade Id of the underlying trade.
state	A	m		Char(4)	Current state of the trade. Valid values are: "ACTI": Trade is active (this is the default value).
contract	A	m		String	Contract code of the trade.
px	A	m		Long	Execution price in currency defined by contract. Value is multiplied by 100, e.g. 1 Euro = 100.
qty	A	m		Integer	Traded quantity.
sellDlvryAreaId	A	m		String	Delivery area of the sell side. Valid value is "CZ".
buyDlvryAreaId	A	m		String	Delivery area of the buy side. Valid value is "CZ".
tradeExecTime	A	m		DateTime	Trade execution time.

Tabulka 22 – Struktura zprávy Public Trade Confirmation Report

2.5.4.10. Contract Information Request (ContractInfoReq)

ContractInfoReq	
Type:	Inquiry Request
Roles:	<ALL>
Routing Keys:	market.request.inquiry
Request Limits:	2/20

Dotaz na kontrakt. Možné se dotazovat max. 7 dní zpětně V případě chybných vstupních parametrů je vrácena odpověď „ErrResp“.

XML Tag	Type	m/o	No.	Data Type	Short description
ContractInfoReq	SE			Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
startDate	A	o		Date	Start date for which the contract information is requested. Notes: - if contract is specified this attribute is ignored - if prodName is specified or neither contract nor prodName are specified, this attribute becomes mandatory.
endDate	A	o		Date	End date for which the contract information is requested. Notes: - if contract is specified this attribute is ignored - if prodName is specified or neither contractId nor prodName are specified, this attribute becomes mandatory.
prodName	CE	o	0..1000	String	The contract information for all contracts belonging to the given products is requested. If prodName is specified, the contract element cannot be specified and the startDate and endDate attributes are mandatory.
contract	CE	o	0..1	String	If contract is specified, the prodName element cannot be specified and the startDate and endDate attributes are ignored.

Tabulka 23 – Struktura zprávy Contract Information Request

2.5.4.11. Contract Information Report (ContractInfoRprt)

ContractInfoRprt	
Type:	Inquiry Response, Broadcast
Response to:	ContractInfoReq (sent to the user-generated private response queue or a broadcast to market . broadcastQueue.<login-id>)
Broadcasted:	Yes
Routing Keys:	<prodName>
Roles:	EmtasImTsAcc, EmtasBalmTsAcc

Informace o kontraktech. Zpráva je distribuována v případě změny atributu na kontraktu nebo jako odpověď na dotaz “Contract Information Request”.

XML Tag	Type	m/o	No.	Data Type	Short description
ContractInfoRprt	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
ContractList	SE	o	0..1	Structure	
Contract	SE	o	0..n		
contract	A	m		Integer	Contract code
prod	A	m		String	Underlying product.
prodRevisionNo	A	m		Long	Revision number of the underlying product.
name	A	m		String	Contract name. This is used for display purposes.
longName	A	m		String	Contract long name, containing additional information.
dlvryStart	A	m		DateTime	Start of delivery.
dlvryEnd	A	m		DateTime	End of delivery.
duration	A	o		Double	The duration of the contract in full hours. For quarterly contracts the

XML Tag	Type	m/o	No.	Data Type	Short description
					value would be 0.25. An hourly contract would have 1.0. An block contract would have value in interval from 2 to 24 (or 23/25 in case of short/long clock change).
predefined	A	m		Boolean	Flag that indicates, if a contract has been automatically created by the system or if the contract was generated with an entry of a user-defined block order. 1 = automatically generated, 0= user defined
state	A	m		String	Current state of the contract. The following values are allowed: "HIBE": Hibernated, the contract was manually deactivated by Central Admin. "ISSUED": The contract is issued, but not available for trading. "OPEN": Contract is active and available for trading. "CLOSE": Contract is closed and not available for trading. "TERM": Contract is terminated and not available for trading. "NOT_ISSD": The contract is not issued and there is not possible to trade on this contract at all.
tradingPhaseStart	A	m		DateTime	Start date and time of the current/next trading phase. When "NOT_ISSD" state is distributed then contains timestamp of the "Not issued" event.
tradingPhaseEnd	A	o		DateTime	End date and time of the current/next trading phase. When "NOT_ISSD" state is distributed then contains timestamp of the "Not issued" event.

Tabulka 24 – Struktura zprávy Contract Information Report

2.5.4.12. Product Information Request (ProdInfoReq)

ProdInfoReq	
Type:	Inquiry Request
Roles:	EmtasImTsAcc, EmtasBalmTsAcc
Routing Keys:	market.request.inquiry
Request Limits:	2/20

Požadavek na detailní informace o produktech.

XML Tag	Type	m/o	No.	Data Type	Short description
ProdInfoReq	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
prodName	CE	o	0..1000	String	

Tabulka 25 – Struktura zprávy Product Information Request

2.5.4.13. Product Information Report (ProdInfoRprt)

ProdInfoResp	
Type:	Inquiry Response, Broadcast
Response to:	ProdInfoReq (sent to the user-generated private response queue or a broadcast to market . broadcastQueue.<login-id>)
Broadcasted:	Yes
Broadcast Routing Keys:	---
Roles:	EmtasImTsAcc, EmtasBalmTsAcc

Detailní informace o produktu jako odpověď na "Product Information Request".

XML Tag	Type	m/o	No.	Data Type	Short description
ProdInfoRprt	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.

XML Tag	Type	m/o	No.	Data Type	Short description
ProdList	SE	o	0..1	Structure	
Prod	SE	o	0..n	Structure	
prodName	A	m		String	Unique identifier name of the product.
dsplName	A	m		String	String used to display the product.
currency	A	m		Char(3)	The currency of the product (e.g. "EUR").
revisionNo	A	m		Long	Revision number of the product. This value is increased by one every time the product is modified by the system.
qtyUnit	A	m		String	Defines the quantity unit.
smallestTradableUnit	A	m		Integer	Defines the smallest tradable unit of the product.
minDsplQty	A	o		Integer	Minimal display quantity.
decShiftQty	A	m		Integer	Decimal shift of the quantity information. A value of 2 results in a display of 100 Kw.
maxQty	A	m		Integer	Maximal allowed quantity for orders entered in contracts belonging to this product.
minPx	A	m		Long	Minimal price allowed for orders entered in contracts belonging to this product.
maxPx	A	m		Long	Maximal price allowed for orders entered in contracts belonging to this product.
decShiftPx	A	m		Integer	Decimal shift of the price information. A value of 2 results in a display in Eurocents.
tickSize	A	m		Integer	Defines the minimum increment for limit prices for this product. The value is entered as an integer, but the decimal price shift is applied.
contractNamePatern	A	o		String	Format string for the contract name.
ProdCfgs	SE	o	0..n	Structure	
cfgKey	A	m		String	Exchange specific product attribute name (currently only blockOrderProduct is supported)
cfgVal	A	m		String	Exchange specific product attribute value. Values for above mentioned keys are true/false

Tabulka 26 – Struktura zprávy Product Information Report

2.5.4.14. Market State Request (MktStateReq)

MktStateReq	
Type:	Inquiry Request
Roles:	EmtasImTsAcc, EmtasBalmTsAcc
Routing Keys:	market.request.inquiry
Request Limits:	1/10

Dotaz na aktuální stav trhu. Požadovaný trh je specifikován v hlavičce zprávy "StandardHeader"

XML Tag	Type	m/o	No.	Data Type	Short description
MktStateReq	SE	m	1	Structure	
StandardHeader	SE			Structure	Standard header of each message. Please see chapter 2.5.1.7.

Tabulka 27 – Struktura zprávy Market State Request

2.5.4.15. Market State Report (MktStateRprt)

MktStateRprt	
Type:	Inquiry Response, Broadcast
Response to:	MktStateReq (sent to the user-generated private response queue or a broadcast to market.broadcastQueue.<login-id>)
Broadcasted:	Yes
Broadcast Routing Keys:	public.<marketId>
Roles:	EmtasImTsAcc, EmtasBalmTsAcc

Aktuální informace o stavu obchodování na trhu. Zpráva je distribuována v případě změny stavu trhu a dále jako odpověď na dotaz "Market State Request".

XML Tag	Type	m/o	No.	Data Type	Short description
MktStateRprt	SE	m	1	Structure	
StandardHeader	SE	m		Structure	Standard header of each message. Please see chapter 2.5.1.7.
state	A	m		Char(4)	Contains the current market state. The following values are allowed: "HIBE": Hibernated; no trading is possible and order books are empty. Done on WebGui by Admin. "ACTI": Market is active and trading is possible.
revisionNo	A	m		Long	Revision number of the market. With every change of the market state this value is increased by one.

Tabulka 28 – Struktura zprávy Market State Report

2.6. Nové scénáře pro stávající způsob automatické komunikace přes komunikační server KSP/KSM

2.6.1. Nastavení/změna/odpověď k novému offline limitu

Aktuální stav offline limitu včetně s ostatními hodnotami bude vracet upravený report aktuálního stavu limitů ve stávající struktuře SFVOTLIMITS.

Nová struktura SFVOTSETTINGS bude sloužit pro nastavení offline limitu přes AK(KSP). Kromě standardní hlavičky a identifikace příjemce a odesílatele bude obsahovat:

SFVOTSETTINGS/Setting – hlavní zapouzdřující datový element

SFVOTSETTINGS/Limit – hlavní element pro nastavení limitu

SFVOTSETTINGS/Limit@type – typ limitu, výčtový typ, nyní jen VDT/VT

SFVOTSETTINGS/Limit@value – nová hodnota pro daný limit v CZK Příklad nastavení limitu na 20tis. CZK:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<SFVOTSETTINGS answer-required="false" date-time="2015-06-24T12:41:08+02:00" dtd-release="1" dtd-version="1" id="123" message-code="475" xmlns="http://www.ote-cr.cz/schema/sfvot/settings">
  <SenderIdentification id="8591824000007" coding-scheme="14"/>
  <ReceiverIdentification id="8591824000007" coding-scheme="14"/>
  <Setting>
    <Limit type="VDT/VT" value="20000"/>
  </Setting>
</SFVOTSETTINGS>
```

Odpověď bude obsahovat strukturu RESPONSE s msg kódem 477 a v případě úspěšného provedení i opis dat v podobě aktuálního stavu limitů (SFVOTLIMITS s msg kódem 476). Budou částečně využity stávající návratové kódy z oblasti finančních reportů:

RESPONSE/Reason@code	Popis
S09000	Požadavek úspěšně zpracován, nastavení změněno.
S09008	Účastník nemá potřebná nastavení (nedefinované limity).
S09009	Na změnu nemá účastník oprávnění.

<i>RESPONSE/Reason@code</i>	<i>Popis</i>
S09010	Nedostatek volných prostředků.
S09011	Neplatná hodnota.
S09012	Neočekávaná chyba..

2.6.2. Zpráva o přesunu části offline limitu do online

Při změně části offline limitu do online, jak je uvedeno v kapitole 8.2.2.2. Vznik obchodu-vše zobchodováno-online FZ zajištění pro stranu B KO, utilizace pokynu v offline < utilizace obchodu v online pro B, bude nutné o tomto stavu účastníka informovat i přes AK. Informace, odeslané na účastníka budou následující:

- Přesunovaná finální částka z VDT limitu do online (Kč) Zbývá částka VDT limitu (Kč)
- Zbývající volné finanční prostředky ve VDT zajištění (Kč) ID obchodu, jež tento přesun vyvolal

Den dodávky obchodu Pro tyto účely bude sloužit nová struktura SFVOTLIMITCHANGE. Bude odesílána nevyžádaně přes KSP. Kromě standardní hlavičky a identifikace příjemce a odesílatele bude obsahovat:

SFVOTLIMITCHANGE/Limits – hlavní zapouzdřující datový element

SFVOTLIMITCHANGE/Limits@trade-date – den dodávky obchodu

SFVOTLIMITCHANGE/Limits@trade-id – id obchodu

SFVOTLIMITCHANGE/Limit – hlavní element limitu

SFVOTLIMITCHANGE/Limit@type – typ limitu, výčtový typ, nyní jen VDT/VT

SFVOTLIMITCHANGE/Limit@value – nová hodnota pro daný limit v CZK

SFVOTLIMITCHANGE/Limit@moved – prostředky přesunuté do jiného typu v CZK (pro VDT/VT do online utilizace krátkodobých obchodů) SFVOTLIMITCHANGE/Limit@free – volné prostředky pro daný limit v CZK

Příklad:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<SFVOTLIMITCHANGE answer-required="false" date-time="2015-06-24T12:41:08+02:00" dtd-release="1" dtd-
version="1" id="123" message-code="478" xmlns="http://www.ote-cr.cz/schema/sfvot/limitchange">
  <SenderIdentification id="8591824000007" coding-scheme="14"/>
  <ReceiverIdentification id="8591824000007" coding-scheme="14"/>
  <Limits trade-id="237445" trade-date="2015-08-31">
    <Limit type="VDT/VT" value="15000" moved="5000" free="1280"/>
  </Limits>
</SFVOTLIMITCHANGE>
```

3. POUŽITÍ ELEKTRONICKÉHO PODPISU

Zprávy jsou předávány mezi klientskou aplikací a backend systémem ve formě XML. Z důvodu zajištění integrity a nepopíratelnosti jsou vybrané zprávy zabezpečeny elektronickým podpisem.

Elektronický podpis je vkládán do následujících zpráv (viz. XSD šablony v kapitole 4)

- OrdModify
- OrdEntry
- ModifyAllOrdrs

Elektronický podpis je vytvářen ve formě Enveloped XML signature (<http://www.ietf.org/rfc/rfc3275.txt>), tzn. Signature element je vložen pod root element XML zprávy a to na konec.

XML podpis **musí obsahovat klientský certifikát** buď ve formě BinarySecurityToken (neboli odkazem v elementu SecurityTokenReference), nebo musí být certifikát zakódován v sekci X509Data. Jiné formy, jako je např. SKI, nejsou podporovány.

3.1. Příklad zprávy s použitím elektronického podpisu

Zpráva před podepsáním

```
<?xml version="1.0" encoding="UTF-8"?>
<OrdModify ordModType="ACTI" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <StandardHeader marketID="IM"/>
  <OrdList>
    <Ord ordId="0" qty="100" revisionNo="0" type="O"/>
  </OrdList>
</OrdModify>
```

bude mít po opatření XML podpisem podobu

```
<?xml version="1.0" encoding="UTF-8"?>
<OrdModify ordModType="ACTI" xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <StandardHeader marketID="IM" />
  <OrdList>
    <Ord ordId="0" qty="100" revisionNo="0" type="O" />
  </OrdList>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod
        Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />
      <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1" />
      <ds:Reference URI="">
        <ds:Transforms>
          <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature" />
        </ds:Transforms>
        <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
        <ds:DigestValue>vx6g0IKv5Qw1nwqOM4hGmn5igXY</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>UJUfISXST2D9FNBah...</ds:SignatureValue>
    <ds:KeyInfo>
      <ds:X509Data>
        <ds:X509Certificate>MIIETCCA32gAwIBAgIDIA+.....</ds:X509Certificate>
      </ds:X509Data>
      <ds:KeyValue>
        <ds:RSAKeyValue>
          <ds:Modulus>xnm5U6RIswp0aRV9ab...</ds:Modulus>
          <ds:Exponent>AQAB</ds:Exponent>
        </ds:RSAKeyValue>
      </ds:KeyValue>
    </ds:KeyInfo>
  </ds:Signature>
</OrdModify>
```

4. ŠABLONY XSD

Šablony XSD jsou uvedené v příloženém souboru: XSD.zip