John Davies
CTO

Incept5

High Performance Financial Service Integration

Tech Mesh
London
5th December 2012

Agenda

- The Problem - Front, Middle and Back office
- Messaging - Volume, Complexity and Value
- MDA, NoSQL, Metadata modelling & Binding
- A few personal tips and questions

Incept5

Financial Services in a Nutshell

- Front Office
  - Traders sit in trading rooms watching screens and talking on telephones (drinking, playing games... oops!)
  - A price hits a trigger, everyone shouts, a deal is agreed and a new trade starts its life cycle
- Middle Office
  - Accountants, Economists and Mathematicians create strategies, watch exposure and risk on client portfolios
  - Trades are matched, cleared and settled against the counter-party and corresponding banks
- Back Office
  - The day's cash movements are aggregated and payment instructions sent out
  - The bank's overnight positions are re-calculated and updated

Incept5

Low Latency

- Front Office - The domain of High Frequency Trading (HFT)
  - Very high volume, from 50k-380k / sec
  - This is per exchange!

  - Latency over 10µS is considered slow
  - 10µS is just 3km in speed of light time!

- Fix is a good standard but binary formats like ITCH, OUCH & OMNet are often better suited

- Much of the data doesn't even hit the processor. FPGA (Field-Programmable Gate Arrays), "smart network cards" do a lot of the work

Incept5

Low Latency - No Java

- We're not going to go here today, it's a world of customised hardware, specialist firmware, assembler and C
- A world where 1ms is estimated to be worth over $100m
  - For that sort of money you program in whatever they want!
  - People who work here are almost super-human, a few make it big but most don't make it at all
- There is little place for Java and VM languages here, we need to move down the stack a little

Incept5

It's not all crazy Low Latency

- Most banks and many brokers do not compete in the low-latency space, their concept of low-latency is measured in milli-seconds
  - The are not arbitrage trading but managing large portfolios

- Fix is far more widely used as a front-office standard in this area, it's still very fast compared to many standards

- Brokers will take feeds from multiple sources (CME, ICE, NYSE, NASDAQ, LSE, LME etc.) and create crossing networks and "dark pools"
  - This is the world of "normal" human beings

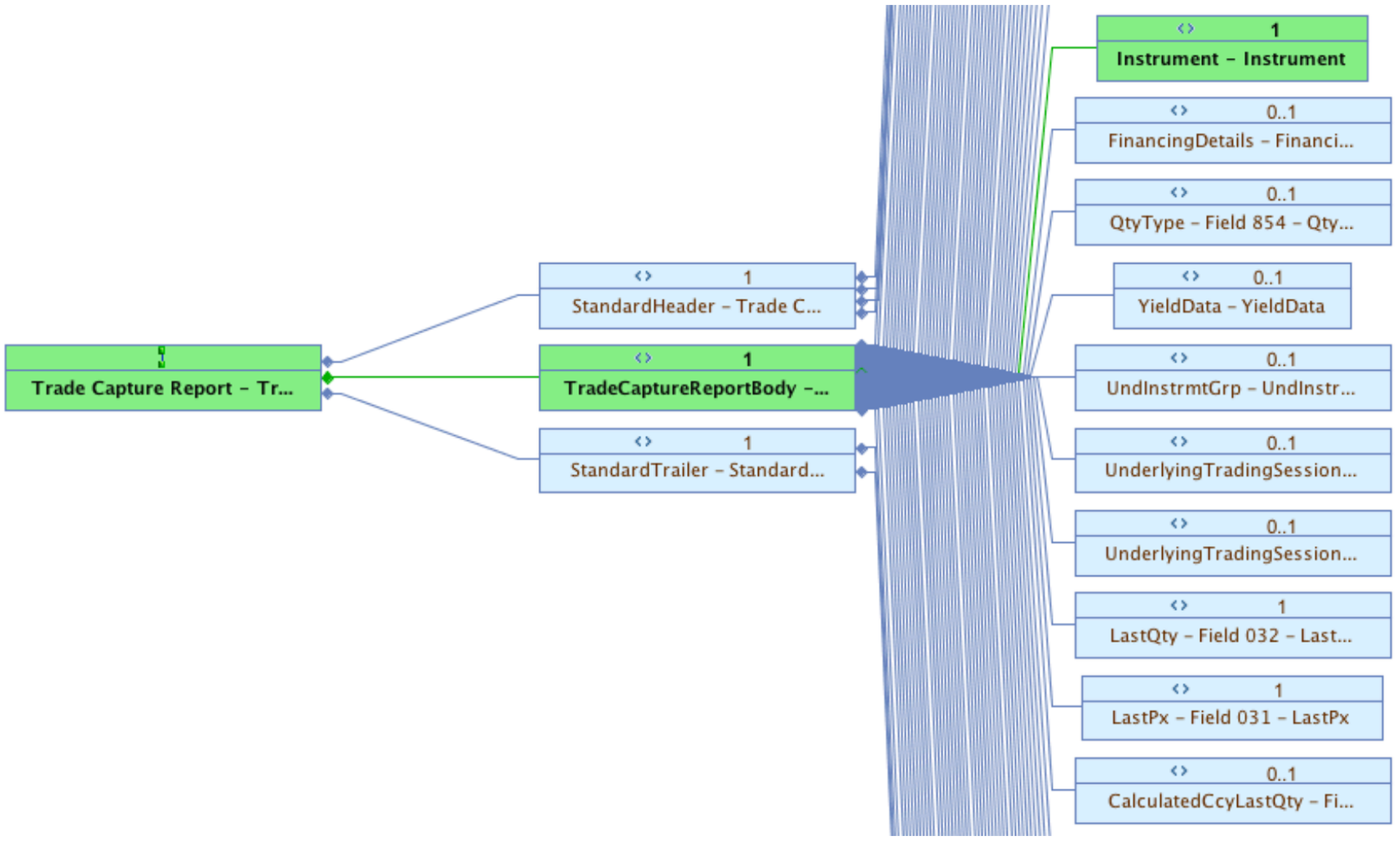
Incept5

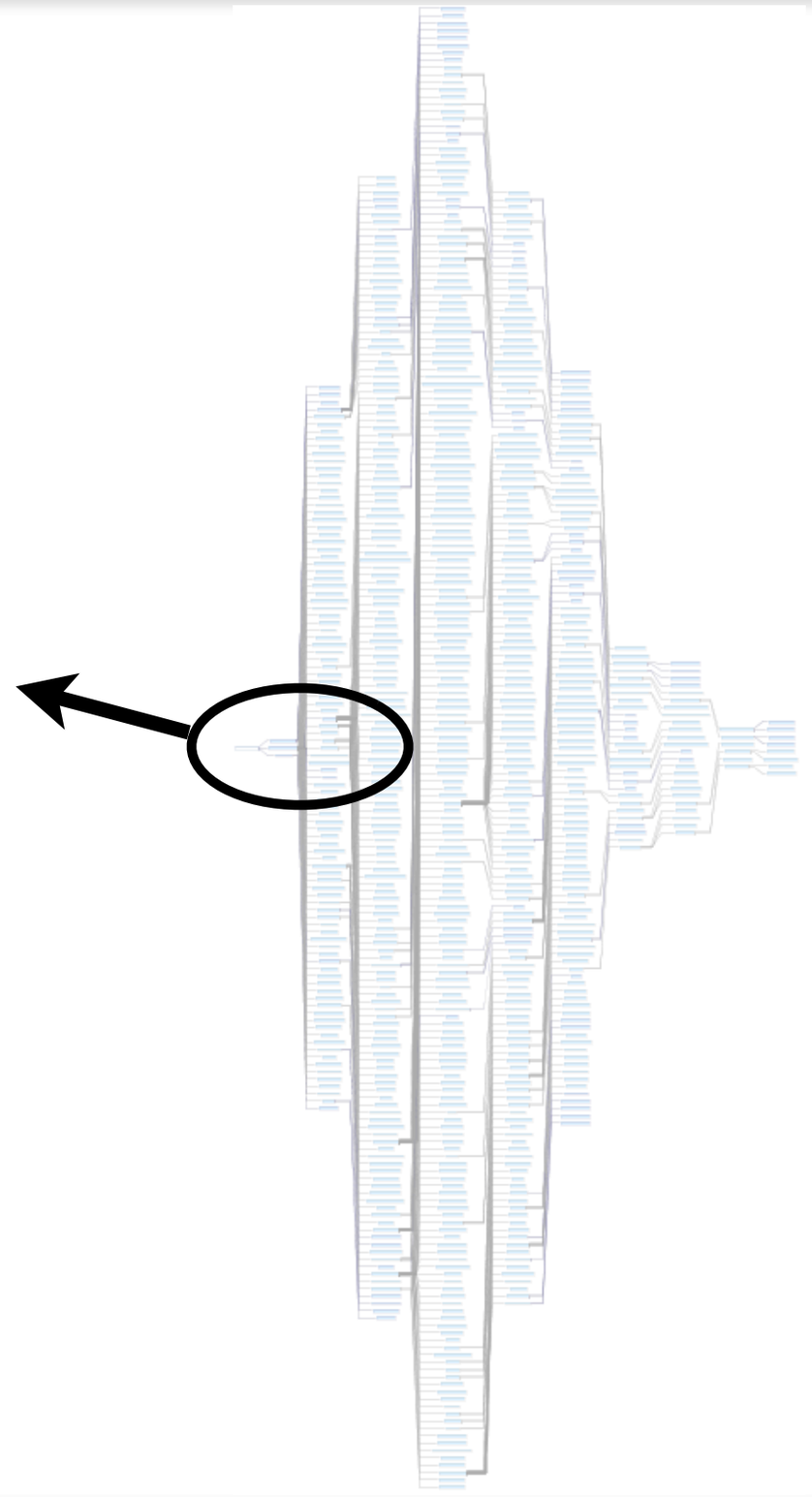
- Good ol' Fix

```
8=FIXT.1.1 9=1 35=AE 1128=7 49=STRING 56=STRING 90=1 91=D 34=1 50=STRING
142=STRING 57=STRING 143=STRING 144=STRING 145=STRING 52=20020101-00:00:00.000
122=20020101-00:00:00.000 212=1 213=D 347=ISO-2022-JP 369=1 627=1 628=STRING
629=20020101-00:00:00.000 630=1 571=STRING 487=0 568=STRING 150=0 572=STRING
17=STRING 527=STRING 378=0 570=Y 55=STRING 65=CD 48=STRING 22=1 454=1 455=STRING
456=1 460=1 461=STRING 167=FAC 200=200201 541=20020101 224=20020101 225=20020101
239=CORP 226=1 227=1 228=1 255=STRING 543=STRING 470=AF 471=GB 472=STRING
240=20020101 202=1 206=0 231=1 223=1 207=XLON 106=STRING 348=1 349=D 107=STRING
350=1 351=D 32=1 31=1 15=AFA 120=AFA 194=1 195=1 30=XLON 75=20020101
60=20020101-00:00:00.000 768=1 769=20020101-00:00:00.000 1033=A 1034=1 1035=E.W
63=0 64=20020101 573=0 574=A1 552=1 54=1 453=1 448=STRING 447=B 452=1 802=1
523=STRING 1=STRING 581=1 81=0 575=Y 576=1 577=0 578=STRING 579=STRING 376=STRING
377=Y 582=1 336=STRING 625=STRING 12=1 13=1 479=AFA 497=Y 157=1 230=20020101 158=1
159=1 238=1 237=1 118=1 119=1 155=1 156=M 77=O 58=STRING 354=1 355=D 518=1 519=1
520=1 521=AFA 136=1 137=1 138=AFA 139=1 37=STRING 198=STRING 11=STRING 38=1 468=0
469=1 528=A 529=1 483=20020101-00:00:00.000 381=1 10=000
```

- It might look relatively simple but it has hidden hierarchy...

Incept5





- A Fix 5.0 SP2 Trade Capture Report

Incept5

The Middle Office

- We're now looking at integration into Fix, some XML, CSVs and the occasional binary format

- We need to load huge amounts of data into large in-memory "databases" or caches (pick your favourite name) to match trades, execute triggers and monitor limits and calculate positions

- We also need to store the data for back-testing and compliance, the volumes are impressive...

Incept5

Storage Volumes

- Take a feed like the CME - 100k/sec is not unusual (recent peek @ 350k/sec)
  - Message size is small, say 512 bytes but that's 50MB/sec or 180GB/hour
  - Not huge but that's just one channel on one exchange, now add two dozen exchanges (one client connects to 28)

- Now we're at 4TB/hour, 36TB/day, 8PB per year etc.

- OK now we're talking but think about parsing this and indexing it at the same time
  - Now it starts to get really interesting

Incept5

On to the Middle Office

- While the front-office volumes are large and high the middle office is somewhat slower and but a lot more complex

- One client portfolio could contain a selection of 50 equities spread over several geographic regions
  - The trading strategy could be to reduce the exposure in one category and increase another, say out of finance into energy across all geographic regions

- You can't just sell one and buy the other, it's done in blocks and allocations, all at different prices each with its own associated risk, some of the risk needs to be balanced

Incept5

Middle Office Messaging - Complex

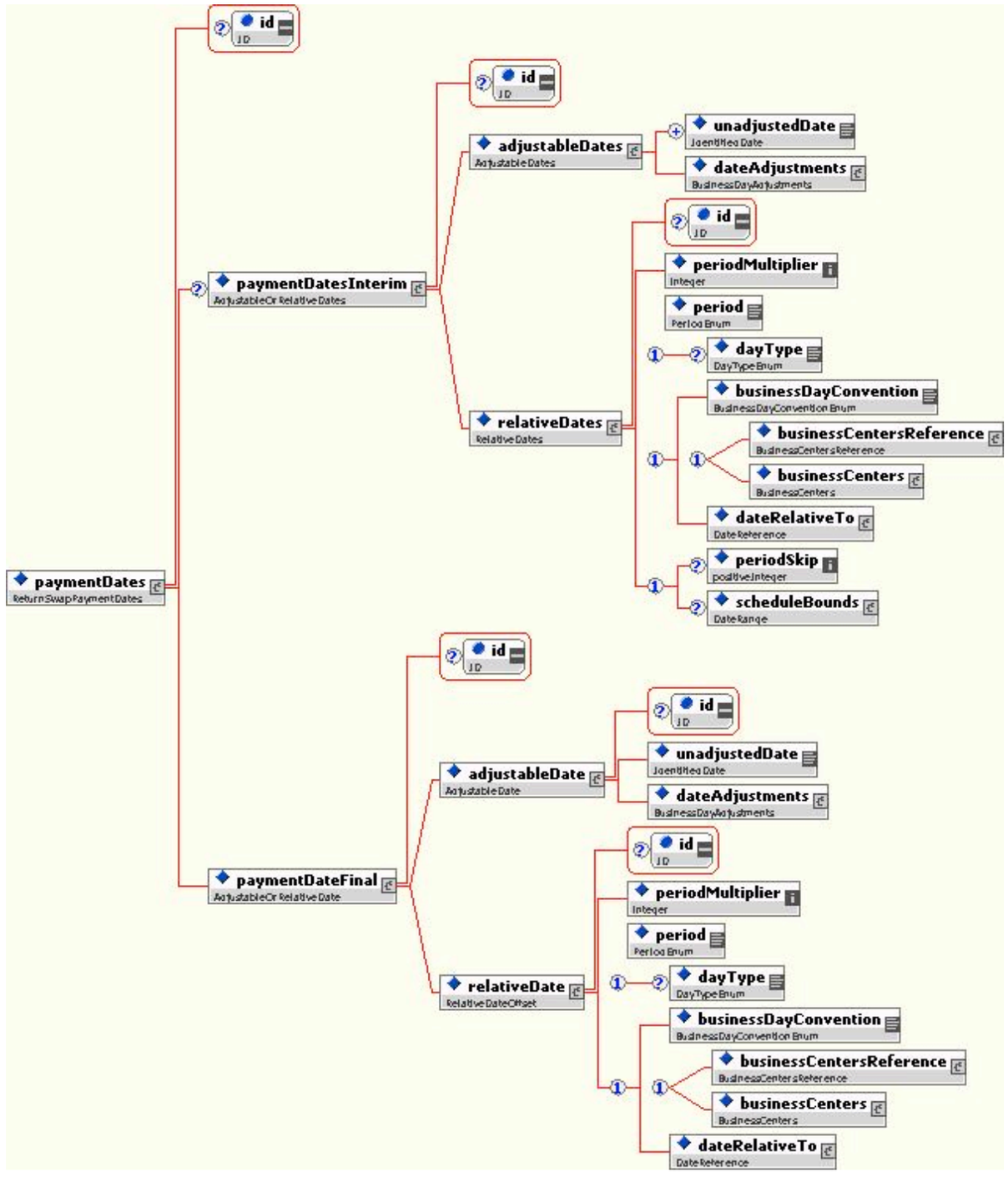
- Volumes are not huge
  - We could see 1000 per second but it's usually lower
- Calculations are complex and grid/HPC is usually required
- Messaging formats are complex
  - Derivative contracts are complex, they are defined in ISDA's FpML
  - Corporate Actions (SWIFT, FpML & XBRL) and now ISO-20022
  - SEPA on ISO-20022, Murex, SwapsWire, CSVs are also common
- XML widely used but usually over MQ & JMS
- Persistence becomes a serious issue

Incept5

FpML

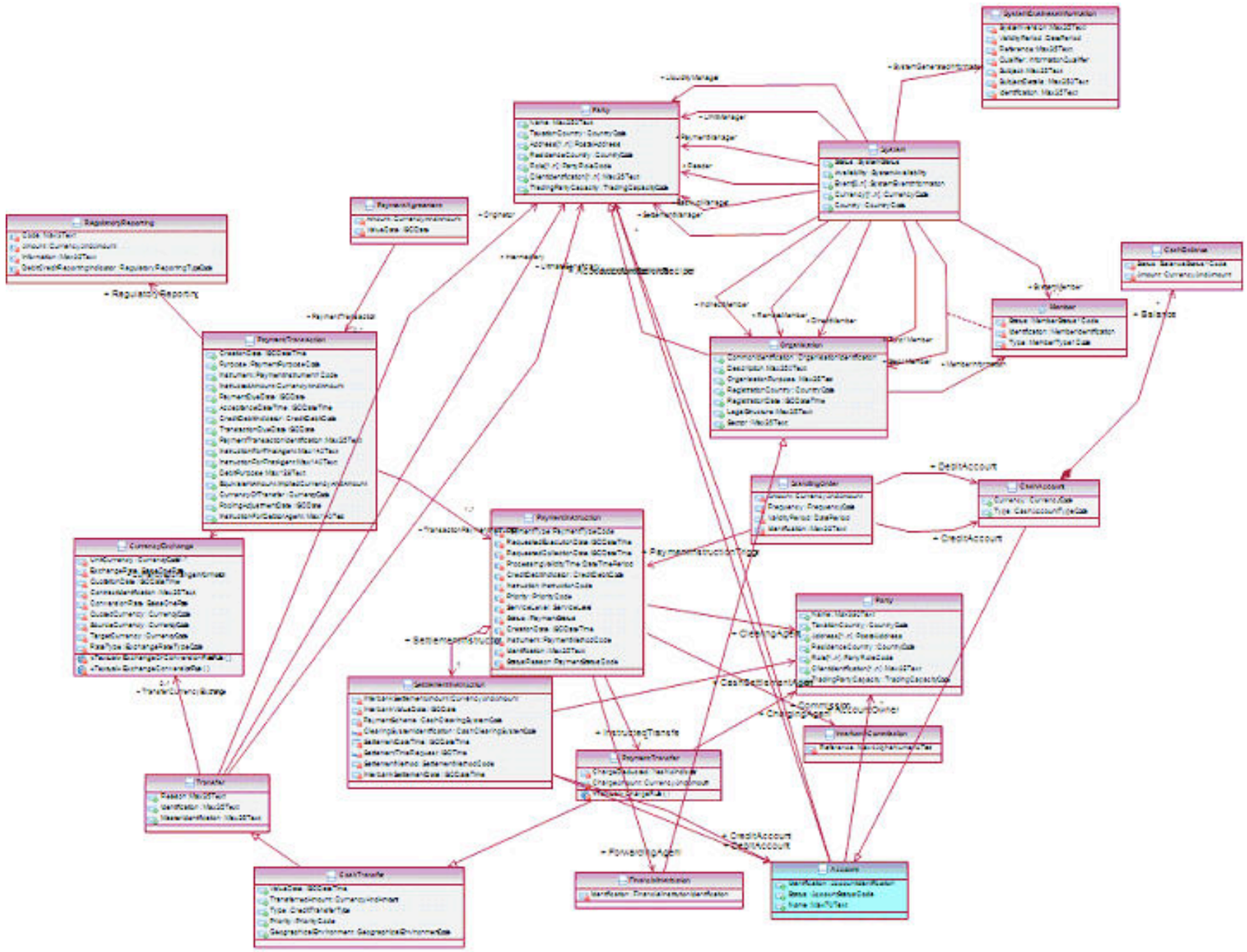
- FpML - Complex
  - 15 levels
  - >3000 elements
  - But well defined

```
<paymentDates id="EquityPaymentDate">
  <paymentDatesInterim id="InterimEquityPaymentDate">
  <relativeDates>
    <periodMultiplier>3</periodMultiplier>
    <period>D</period>
    <dayType>CurrencyBusiness</dayType>
    <businessDayConvention>FOLLOWING</businessDayConvention>
    <businessCenters id="PrimaryBusinessCenter">
      <businessCenter>USNY</businessCenter>
    </businessCenters>
    <dateRelativeTo href="InterimValuationDate"/>
  </relativeDates>
 </paymentDatesInterim>
 <paymentDateFinal id="FinalEquityPaymentDate">
  <relativeDate>
    <periodMultiplier>3</periodMultiplier>
    <period>D</period>
    <dayType>CurrencyBusiness</dayType>
    <businessDayConvention>FOLLOWING</businessDayConvention>
    <businessCentersReference href="PrimaryBusinessCenter"/>
    <dateRelativeTo href="FinalValuationDate"/>
  </relativeDate>
 </paymentDateFinal>
</paymentDates>
```



ISO-20022 & SEPA

- The new standard for standards, ISO-20022
  - The foundation for SEPA (Single European Payments Area)



Implement this!

- From SEPA
  - ChequeMaturityDateRule: If ChequeType is present and is DRFT or ELDR, then ChequeMaturityDate is optional. If ChequeType is not present or is different from DRFT or ELDR, then ChequeMaturityDate is not allowed.

- From FpML
  - ird-57 (Mandatory): If rollConvention is neither NONE nor SFE, nor a day of the week (MON, TUE, WED, THU, FRI, SAT or SUN) then the period must be M or Y.

Incept5

Integration - High Value

- Back Office
  - Low volume (10-1000 / hour), sometimes just a few per day
  - Very high value messages, strict compliance and validation
  - Often several $trillion per day
  - Proprietary networks, mostly SWIFT

- Back to numbers...
  - At 1% interest per year that's 0.0027% per day.
  - 0.0027% of $1bn is $27,261.55
  - That's $1m interest per day for each $37bn
  - Today USD/GBP went from 1.60 to 1.61, that's 0.625% (not unusual)
  - The loss (or gain) from 0.625% in one day is $6m per $billion!

Incept5

SWIFT - MT564 Corporate Action Notification

- Plenty of time for the "<" and ">" but it's 30 years old and 9,000 banks already use it

{1:F01INTRUS33AXXX9999999999}
{2:O564094704012 7FRNYUS33AXXX4218183425040127094 7N}{3:
{108:MT564}}{4:

:16R:GENL

:20C::SEME//20030418000000 42

:20C::CORP//12345

:23G:NEWM/CODU

:22F::CAEV//XMET

:22F::CAMV//VOLU

:98A::PREP//20010901

:25D::PROC//PREC

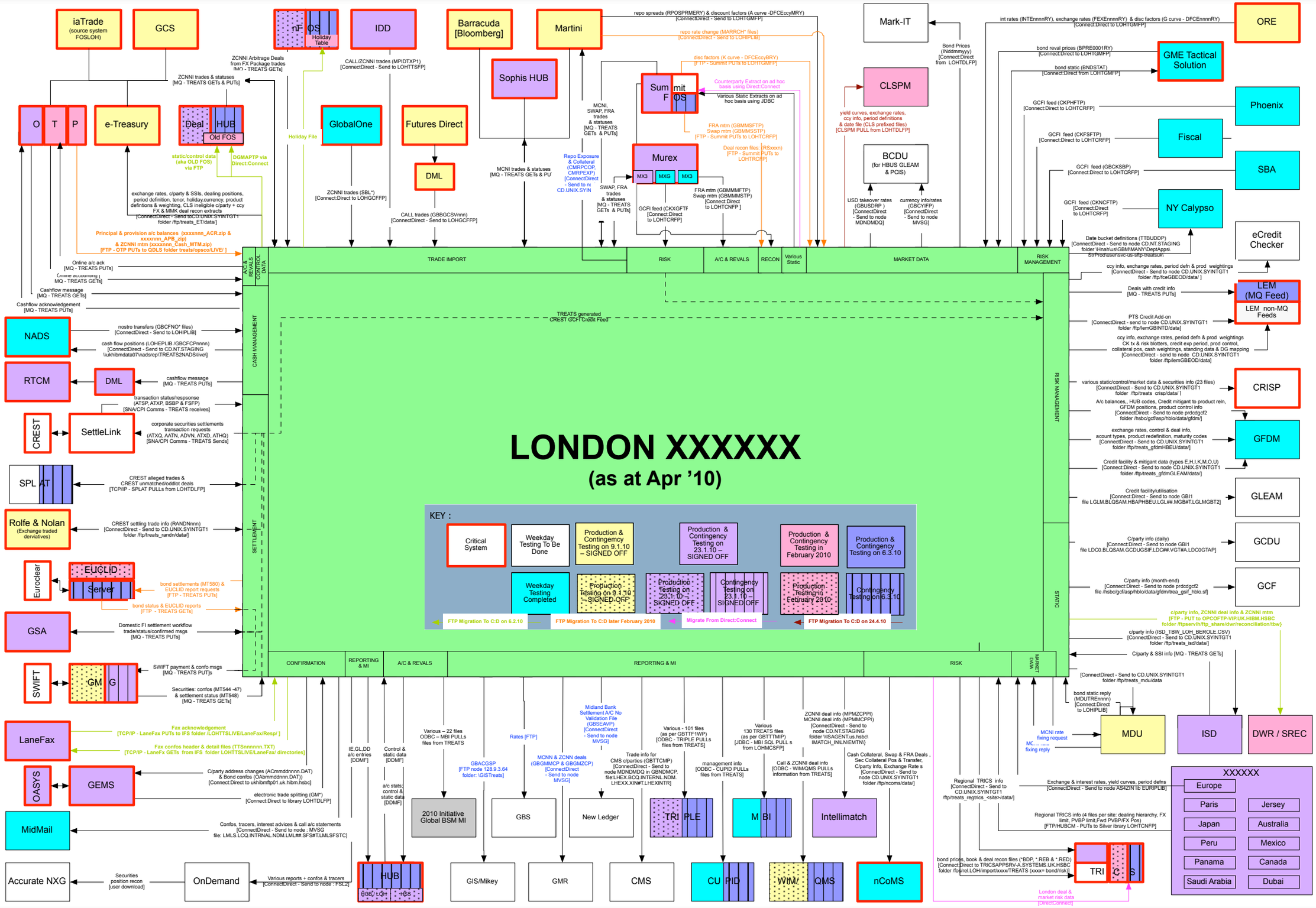
OK So it's a REALLY Nasty!

- OK, OK, we get it, it's serious volumes, really fast, horrendously complex and we can't afford to make a mistake

- So where do the banks use all this stuff?

Incept5

System Architecture View



Functional Landscape

Front Office

- Sales
- Trading
- Client Relationship Management
- Marketing & Advertising
- Product Development
- Research

Middle Office

- Order/Trade Matching
- Trade Allocation & Booking
- Trade Desk Support
- Pricing & Mark-to-Market
- Prime Brokerage Support
- Risk Control

Back Office Operations, Services & Control

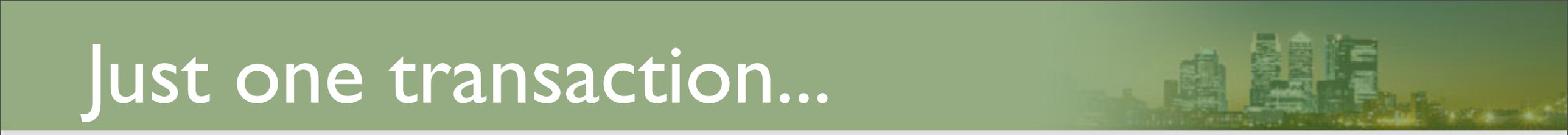
- Confirmation Generation
- Confirmation Matching
- Confirmation Exception Management
- Interface Management
- Priority/Position Management
- Queue Management
- Exception Management
- Fail Control
- Funding
- Payments
- Billing/ Brokerage Fees
- Post-Settlement Reconciliation
- Collateral Management
- Margin
- Securities Borrowing & Lending
- Asset Servicing
- Safekeeping/ Custody
- Reference Data
- Interest Claims Management
- Operations Control
- Regulatory & Compliance
- MIS
- Complex Products
- Process Design
- Process Management
- Client Services

Finance

- Treasury
- Accounting
- Management Reporting
- Regulatory Reporting
- Strategic Planning
- Budgeting & Forecasting
- Product Control
- Tax

Incept5

Just one transaction...



Front Office

- Sales
- Trading
- Client Relationship Management
- Marketing & Advertising
- Product Development
- Research

Middle Office

- Order/Trade Matching
- Trade Allocation & Booking
- Trade Desk Support
- Pricing & Mark-to-Market
- Prime Brokerage Support
- Risk Control

Back Office Operations, Services & Control

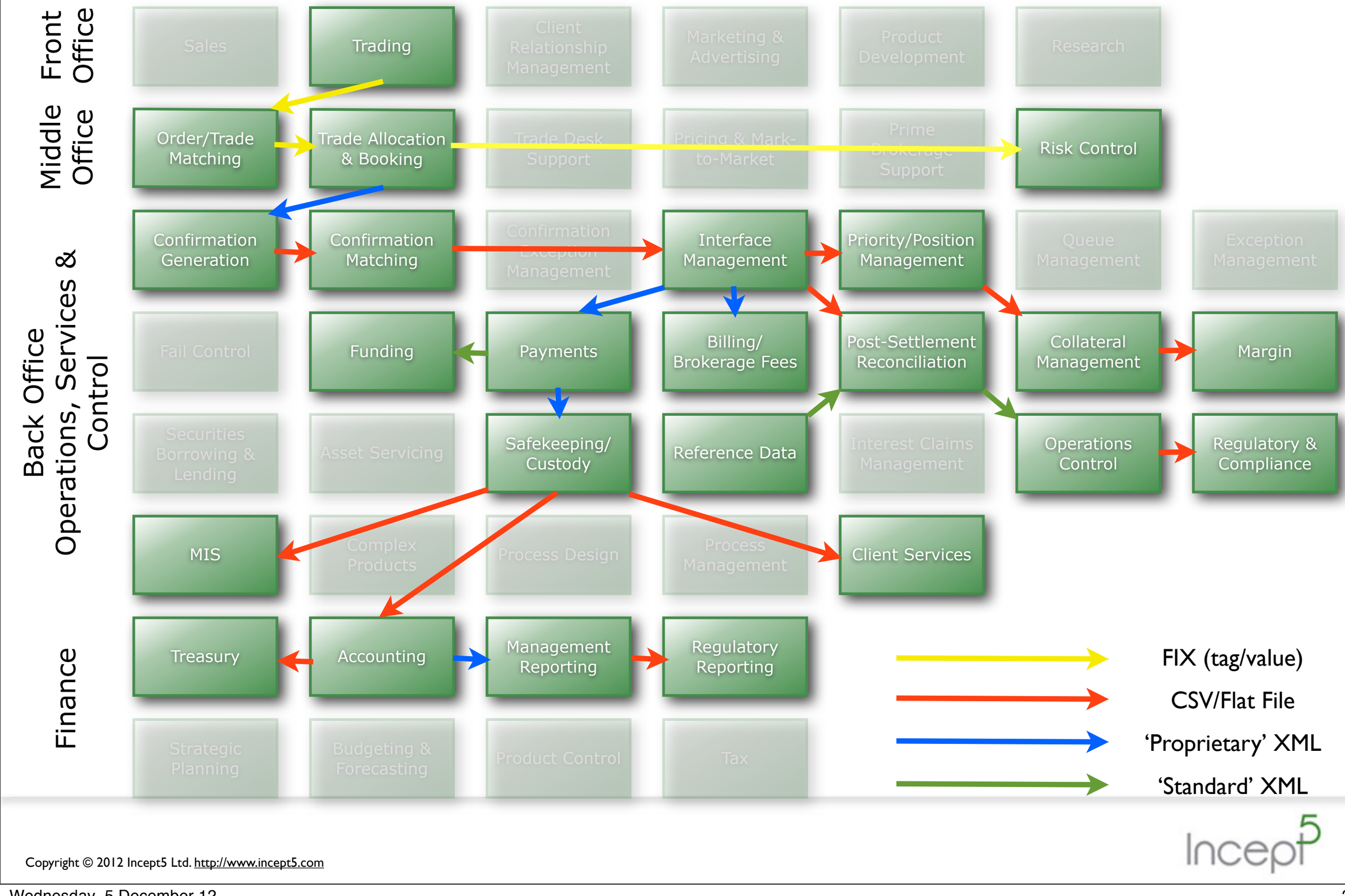
- Confirmation Generation
- Confirmation Matching
- Confirmation Exception Management
- Interface Management
- Priority/Position Management
- Queue Management
- Exception Management
- Fail Control
- Funding
- Payments
- Billing/ Brokerage Fees
- Post-Settlement Reconciliation
- Collateral Management
- Margin
- Securities Borrowing & Lending
- Asset Servicing
- Safekeeping/ Custody
- Reference Data
- Interest Claims Management
- Operations Control
- Regulatory & Compliance
- MIS
- Complex Products
- Process Design
- Process Management
- Client Services

Finance

- Treasury
- Accounting
- Management Reporting
- Regulatory Reporting
- Strategic Planning
- Budgeting & Forecasting
- Product Control
- Tax

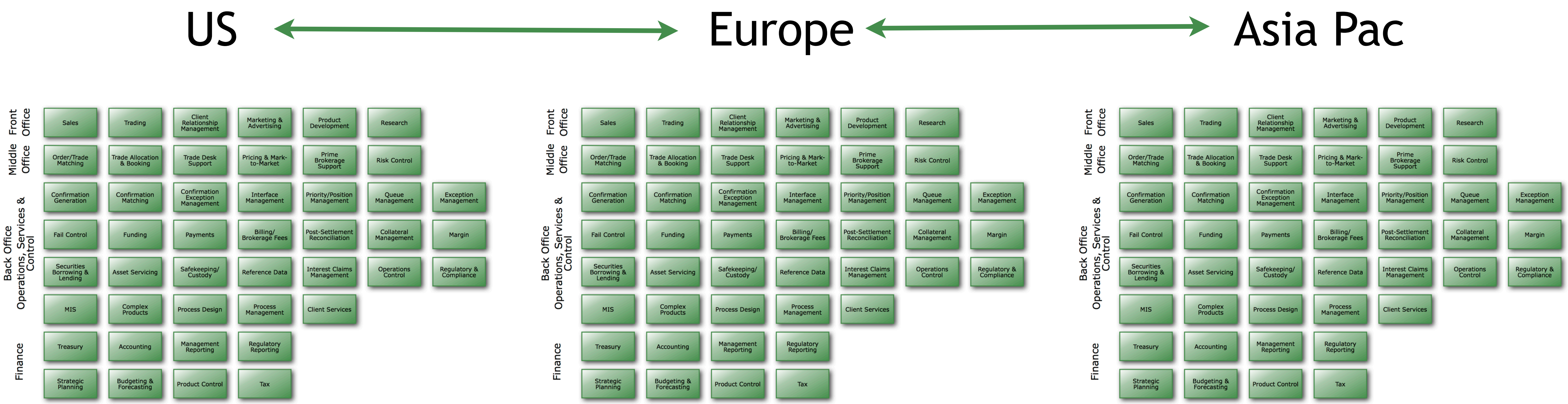
Incept5

Just one transaction...



Incept5

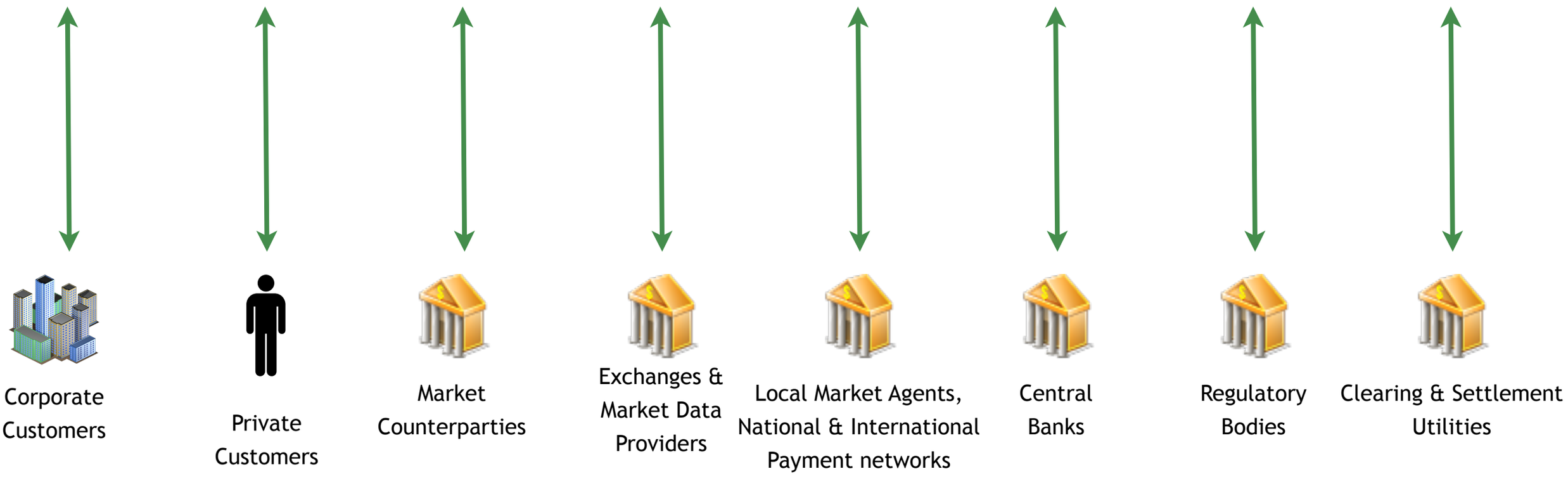
Now add the geography...



Incept5

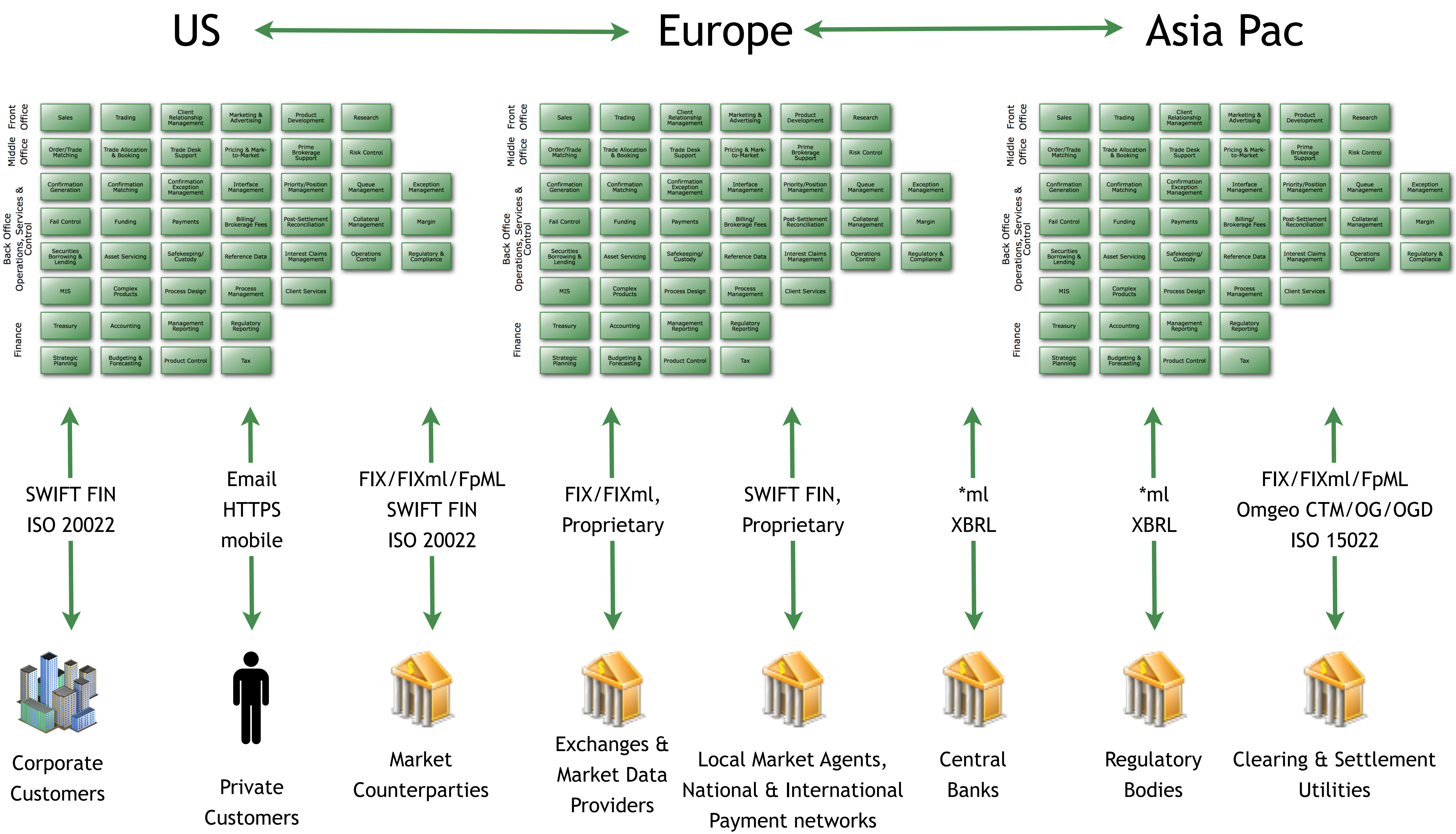
And the rest of the world...

US ⟷ Europe ⟷ Asia Pac



Incept5

Add a few standards...



Incept5

Model Driven Architecture

- We model the data, like a database schema or XML schema
- Our front office trade is part of the middle office derivative and forms part of the back office settlement
- Although the format changes as we move through the bank the basic meta-data does not
  - A payment is a payment
  - An amount is an amount
- From the meta-data we can generate code - In our case Java

Incept5

CSV & Text files

- One of the simplest but also one of the most common

```
Name,Card Number,Expiry Date,Amount,Currency,Transaction Date,Commission
Oliver Twist,4325-6486-3757-2678,10/06,100,GBP,26-09-2006,8,14988603,UK
Uriah Heep,4724-7345-4725-7835,11/04,258,USD,21-12-2006,5,15688632,UK
Mr Scrooge,4924-7264-1264-8536,04/09,1250.6,USD,13-09-2006,15,66846035,US
Charles Dickens,4457-436-0087-0104,05/08,350,EUR,13-11-2006,10,93440252,DE
RowCount=4
```

- While any programmer can parse this in minutes it's not obvious how to...
  - Validate semantically & syntactically
  - Route based on values
  - Transform into another format
  - Manage multiple similar formats in a consistent way
  - Document the format / contract

Incept5

FIX isn't complex

- FIX is "very" simple, it's basically tag/value pairs

```
8=FIX.4.1 9=154 35=6 49=BRKR 56=INVMGR 34=236
52=19980604-07:58:48 23=115685 28=N 55=SPMI.MI
54=22 7=200000 44=10100.000000 25=H 10=159
```

- So, simple, the tag represents the field...
  - 44 refers to Price
  - 52 is sending Date/Time
  - 55 refers to the symbol

- Basic but it's still better than XML when latency comes into play

Incept5

SWIFT - MT564 Corporate Action Notification

- Plenty of time for the "<" and ">" but it's 30 years old and 9,000 banks already use it

{1:F01INTRUS33AXXX9999999999}
{2:O564094704012 7FRNYUS33AXXX4218183425040127094 7N}{3:
{108:MT564}}{4:

:16R:GENL

:20C::SEME//20030418000000 42

:20C::CORP//12345

:23G:NEWM/CODU

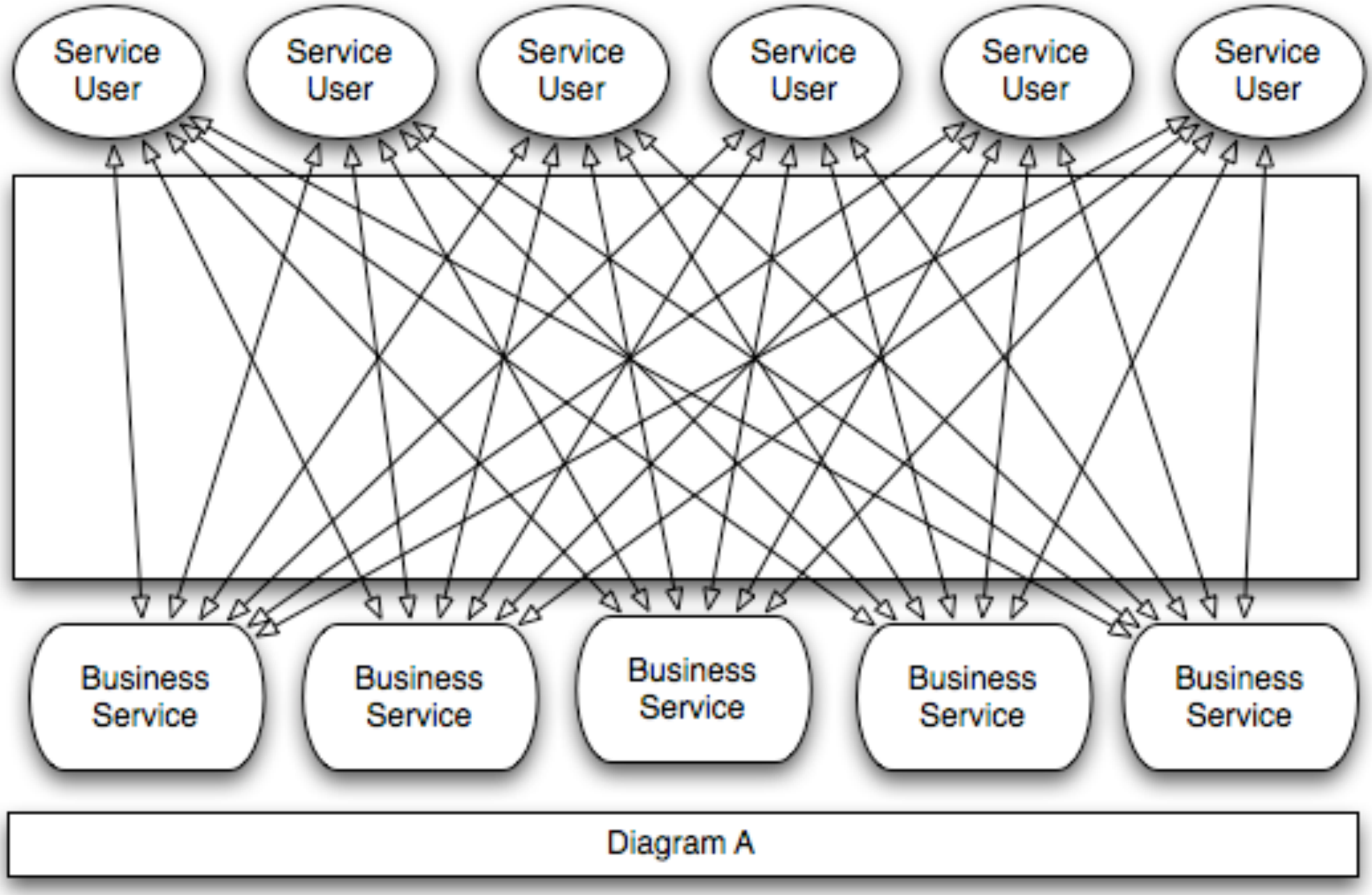
:22F::CAEV//XMET

:22F::CAMV//VOLU

:98A::PREP//20010901

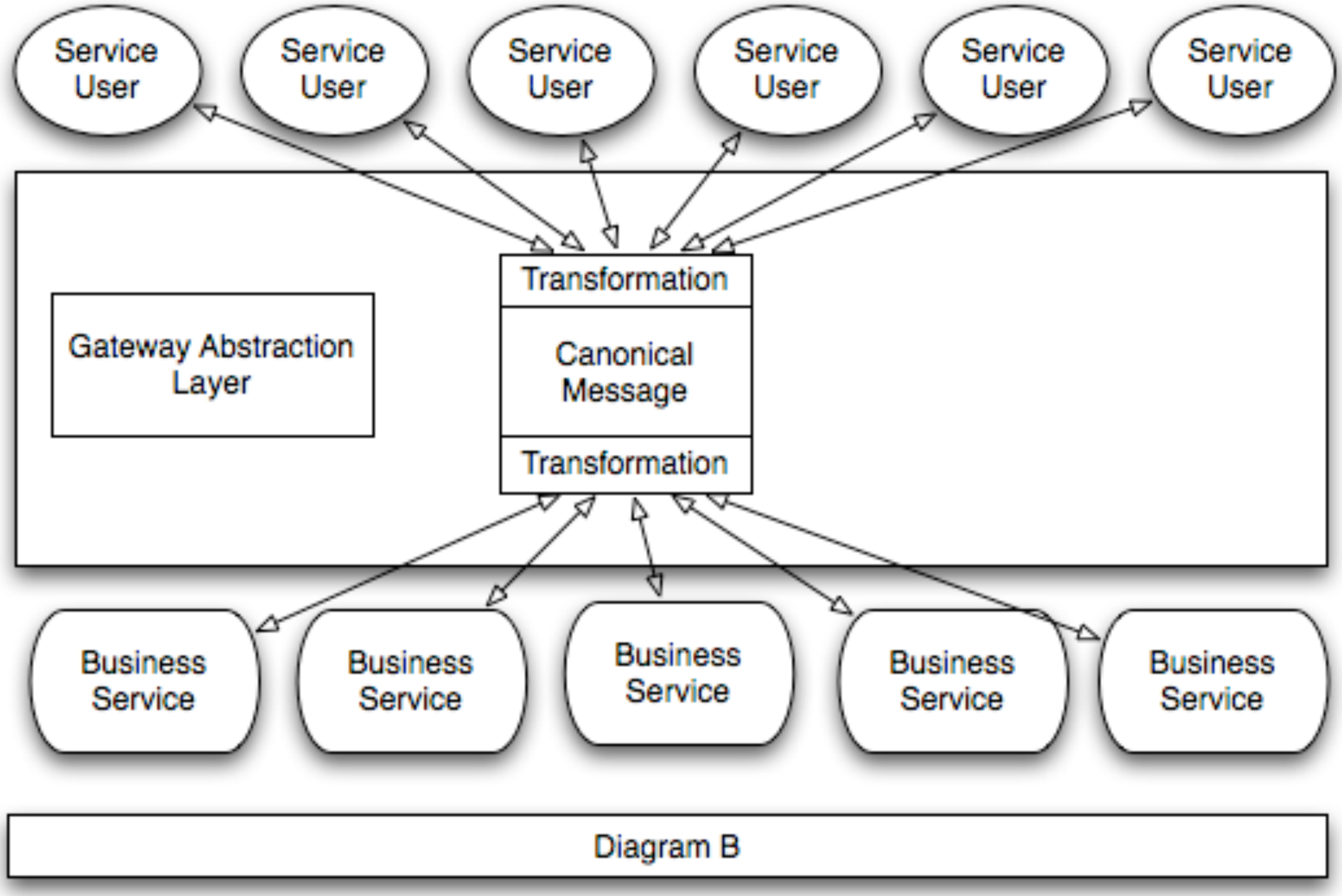
:25D::PROC//PREC

How not to do it



Incept5

A good idea but bad in practice



Incept5

Metadata modelling

ISO-20022 (or FpML) Standard

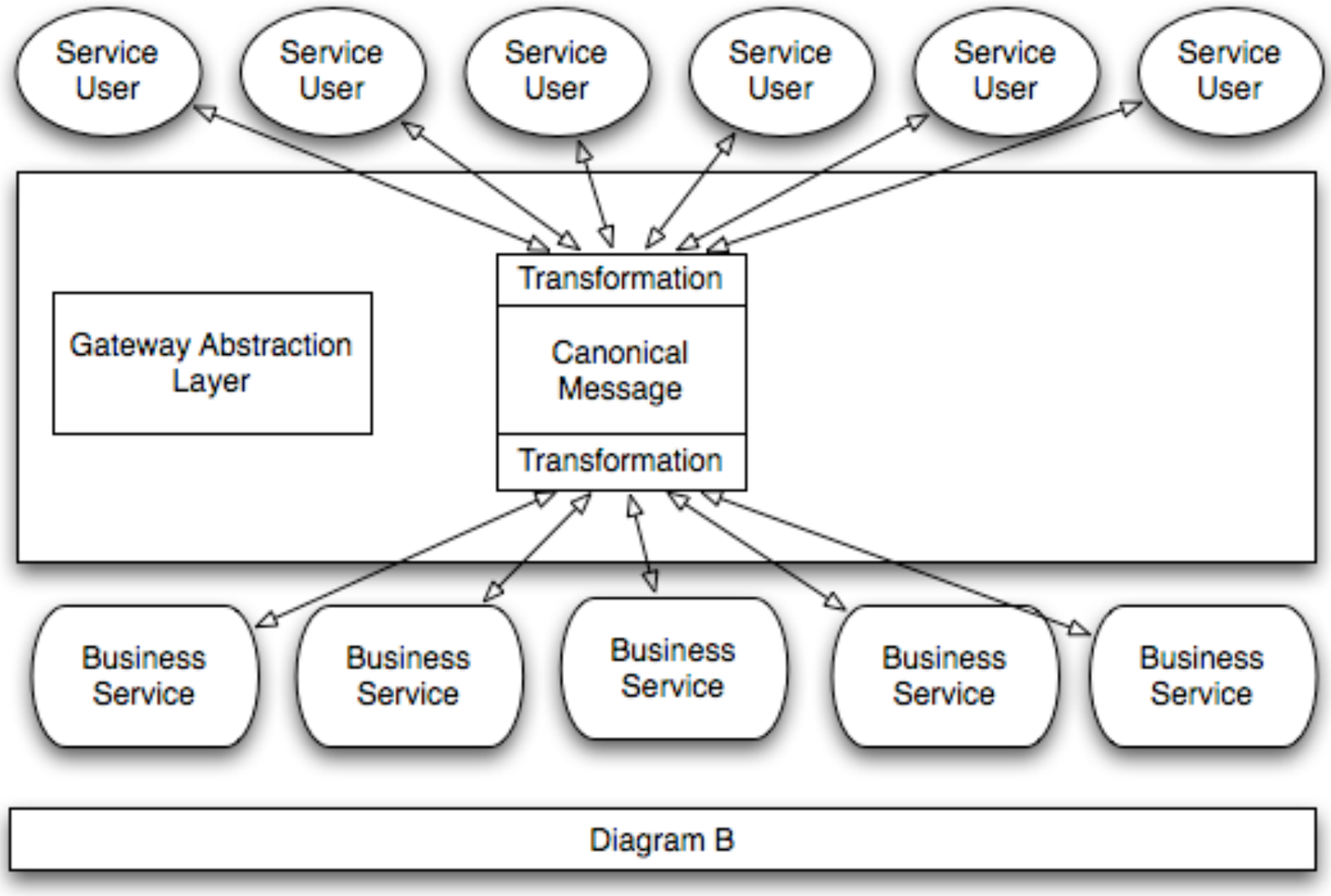
Payment Date

```
8=FIX.4.1 9=154 35=6 49=BRKR
56=INVMGR 34=236
52=19980604-07:58:48
23=115685 28=N 55=SPMI.MI
54=22 7=200000
44=10100.000000 25=H 10=159
```

{2:O564094704012 7FRNYUS33AX
XX4218183425040127 0947N}{3:
{108:MT564}}{4:
:16R:GENL
:20C::SEME//20030418000000 42
:20C::CORP//12345
:23G:NEWM/CODU

We can achieve the same effect

- If we model the messages to something like ISO-20022 or FpML then we can achieve a similar architecture without having to map everything to a canonical model



A solution

- We model every message and end-point and in most cases use Java Binding technologies

- We map message elements to a meta-data standard
  - Often ISO-20022/SEPA or something like FpML

- We now have self-validating objects that an auto-map many/most of their elements to any other message

- We can retain every message in it's original format/structure so versioning ceases to be an issue
  - Messages can be persisted in their original format

Incept5

Persistence

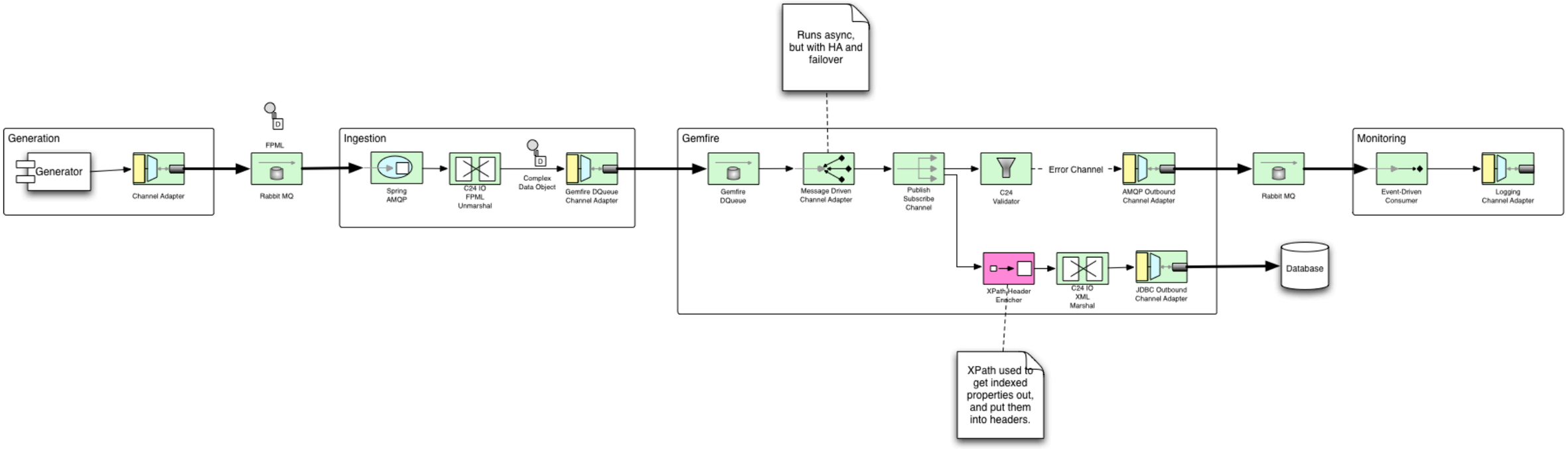
- How do you store something as complex as FpML or SEPA's ISO-20022 messages in a relational database?
  - FpML is typical, it has over 1000 elements and umpteen levels of depth
  - Normalising FpML would result in the mother of all databases and SQL queries up to a page long
- How do you manage multiple versions?

- Answer
  - Don't use a relational mapping
  - Simply store it as XML (in a CLOB) and extract the indices you need with XPath
  - Many databases (Oracle, Sybase, DB2 V9 etc.) offer XML data types but they usually don't implement all the schema features and slow the insert times down to a crawl
  - Or use a "NoSQL" solution like MongoDB or Redis

Incept5

Getting on with Business

- The business requirements define workflow, not how to implement the integration
- Should we really care what the message type is or it's format?
- The workflow should be the same regardless of the messages we're processing



In no Particular Order...

- KISS
- No ORM!!!
- Distributed Architecture
- MDA - Java-Binding (we used C24's Integration Objects)
- Actor-based Concurrency - AKKA
- Test, code, test, code...
- No Transactions!
  - Compensational Transactions
- Multiple Languages (DSLs)
- Oh yes, no ORM!

Incept5

It's question time...

- John.Davies@Incept5.com
- Twitter: @jtdavies

Incept5