

-
-
-
-
-
-
-
-
-
-

How conceptual system architecture leads to business processes

Hernan Astudillo

Soluciones S.A.

hernan@acm.org

www.ime.usp.br/~ha/slides/oopsla2k.ppt

-
-
-
-
-
-
-
-

-
-
-
-
-
-
-
-
-
-

Context



-
-
-
-
-
-
-
-

-
-
-

Project context

- Allow payments from Financial Institutions (FI) to various Federal Agencies
- In actuality, these payments are
 - ...collected by the Treasury
 - ...on behalf of the Agencies
 - ...and reported to the Federal Reserve Banks (FRBs)
- FIs may be anywhere in the world
- Agencies are located mostly in D.C. area
- 37 FRBs located across the U.S.

•
•
•

Project goals: functional

- Deposit reporting
- Accounting classification for cash management purposes
- Reconciliation tools for client agencies
- Consolidated financial reporting
- Cash forecasting (deposits & disbursements)

-
-
-

Project goals: non-functional

- Consolidated, standardized information
 - all-electronic
 - open architecture
 - single on-line stream to agencies
- Simplified access to information
 - in format (standardized) and platform (web)
- Robust and adaptable

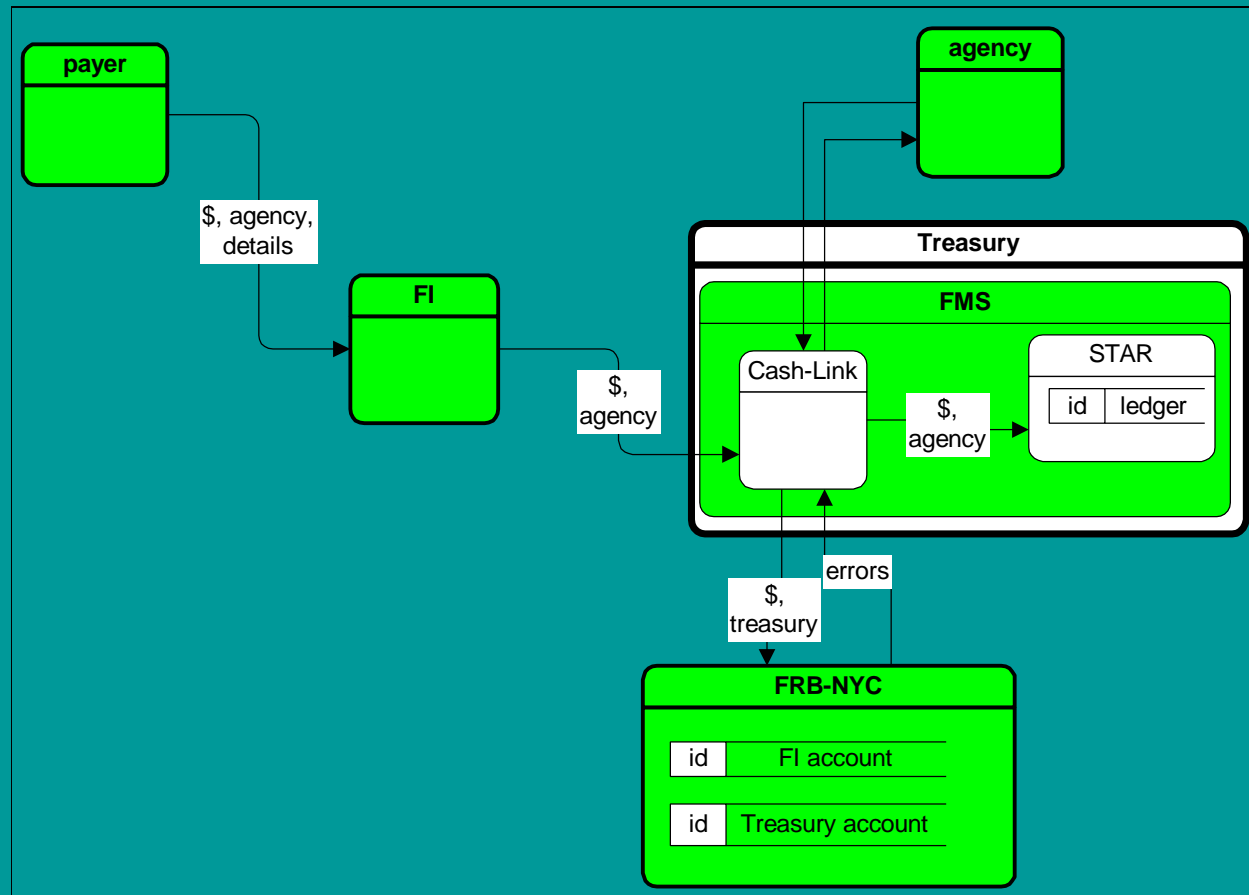
-
-
-
-
-
-
-
-
-
-

On discovering



-
-
-
-
-
-
-
-

Relevant business processes



2000-10-18

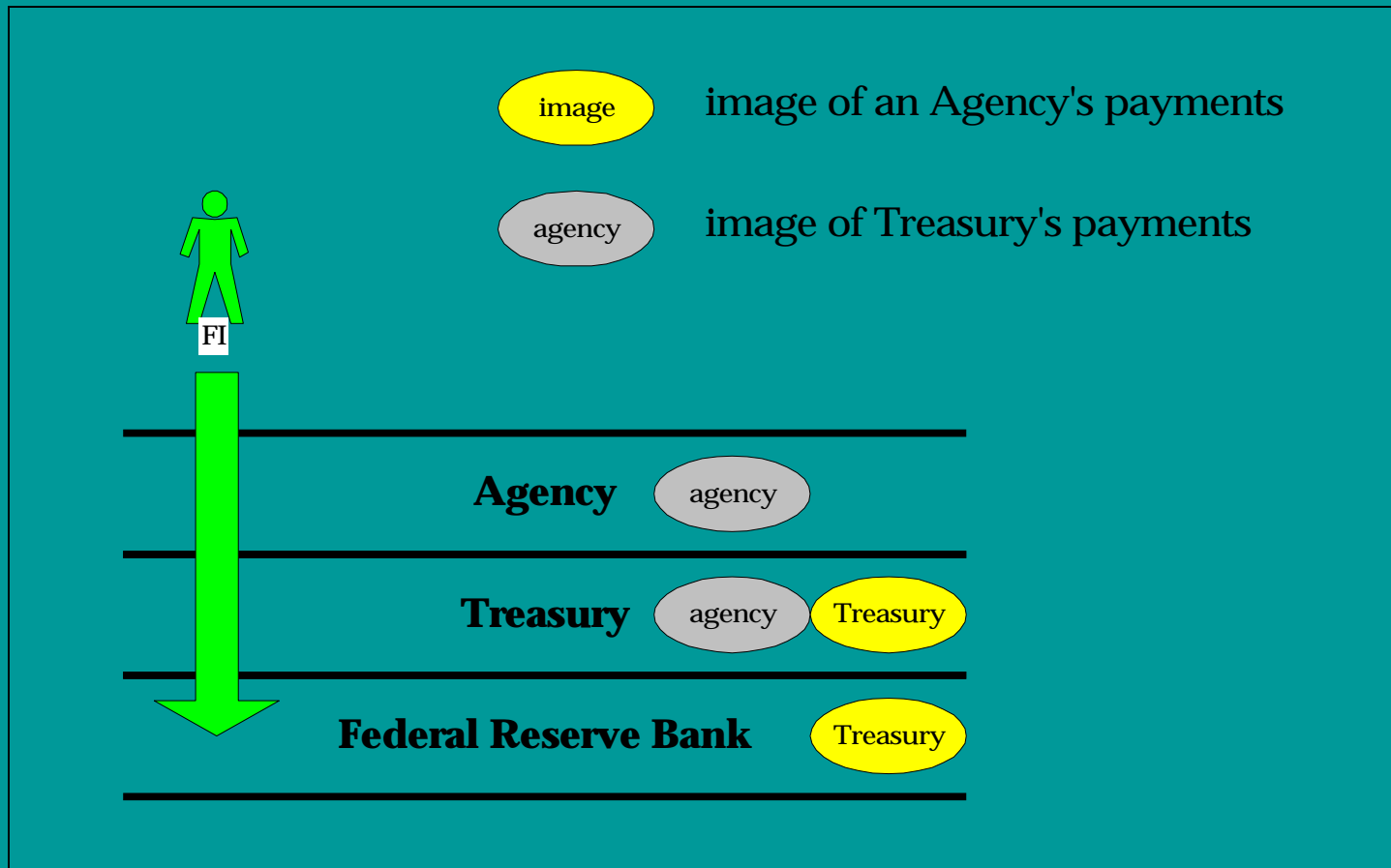
Astudillo / OOPSLA'2000

-
-
-

Reasoning

- System allows:
 - FI s pay to Agencies
- Well, not really:
 - Treasury collects and keeps this payments
- Well, not really:
 - fund transfers happen *inside* the FRBs
- So we support an accounting fiction
- ...but this doesn't help to understand reconciliation and forecasting

Key representation

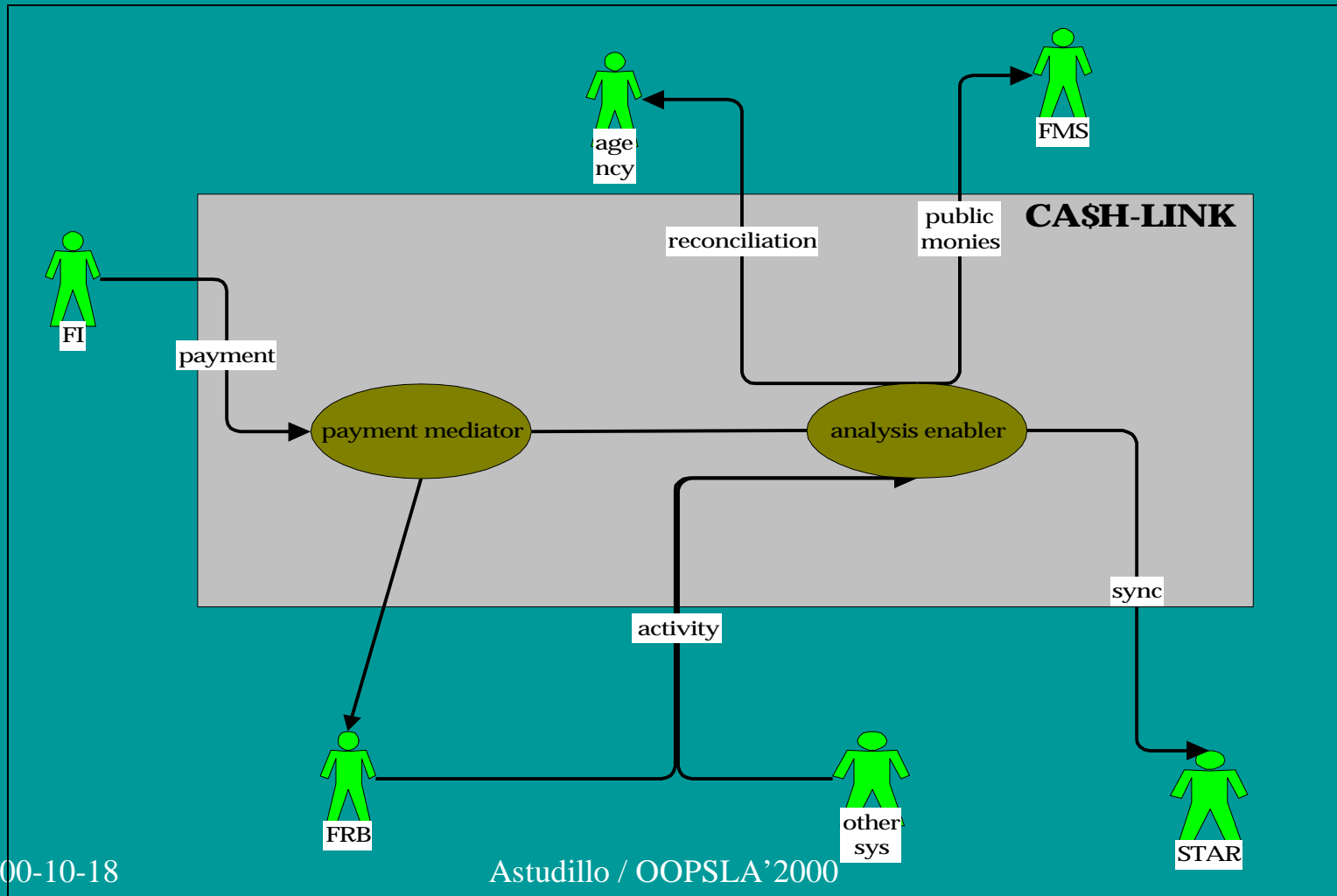


-
-
-

Key insight

- Paying and reflecting are architecturally different
 - OLTP, available, real-time, geographically replicated vs. datahouse, 9-to-5, batch, central location
- Payment processing ? post-facto analysis
- Hence, two system halves:
 - “payments mediator”
 - “analysis enabler”
- Business processes!!
 - (so sayd the analysts)

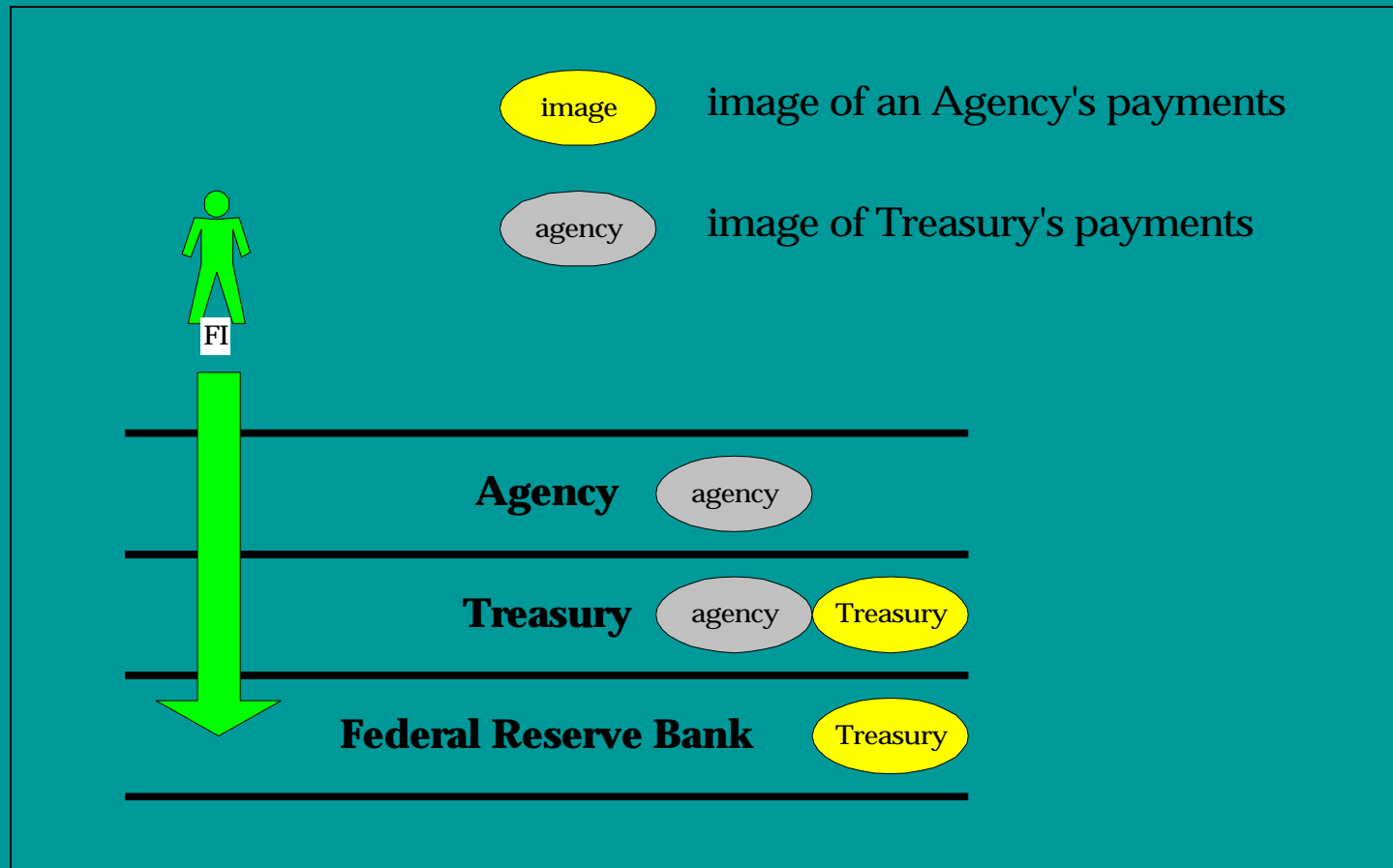
The two "lobes"



2000-10-18

Astudillo / OOPSLA'2000

Key representation



-
-
-

“Lobes”?

- Lobe:
 - aggregate of business functions demanding similar mechanisms across several dimensions (e.g. availability, distribution, interactivity...)
- “This system is composed of two lobes”...
- ...or rather:
 - “this system’s overall functionality can be meaningfully aggregated into two large sets with meaningful architectural differences”

-
-
-

Characteristics of the lobes

Characteristic	Payment mediator	Analysis enabler
Timeliness	Near-time processing	Batch processing
Typical transactions	OLTP-like	DSS-like
Synchronicity	Event-driven	Data-driven
Availability	24x7	Daily
Location	Distributed, fail-over	Centralized (internal replication)
Reliability	Extremely high (\$)	Plain old high

-
-
-

Lobes architecture

- Payment mediator
 - fully distributed, fully replicated, fail-over replicated, supporting only payment processing
 - all historical analysis (i.e. large volumes of data) *must* be left to analysis enabler
- Analysis enabler
 - centralized, internally replicated (for load-balancing and availability), data-centric architecture
 - all near-time payment processing *must* be left to payment mediator

-
-
-
-
-
-
-
-
-
-

On documenting



-
-
-
-
-
-
-
-

-
-
-

Architectural strategy

- Strategy: “multi-layered derivation”
- Aspects:
 - Refinement
 - different abstraction and completeness levels
 - Layered description
 - diagrams and entities for each layer
 - Traceable refinement
 - traceability matrices

-
-
-

The 5 layers (1/2)

- 1) Lobes
 - aggregate business functions that demand similar mechanisms
- 2) Components
 - large-grain software or data pieces
 - each component offers “services” to manipulate what is conceptually a single unit
 - typical component kinds: lifecycle, reporting, importing [details later]

-
-
-

The 5 layers (2/2)

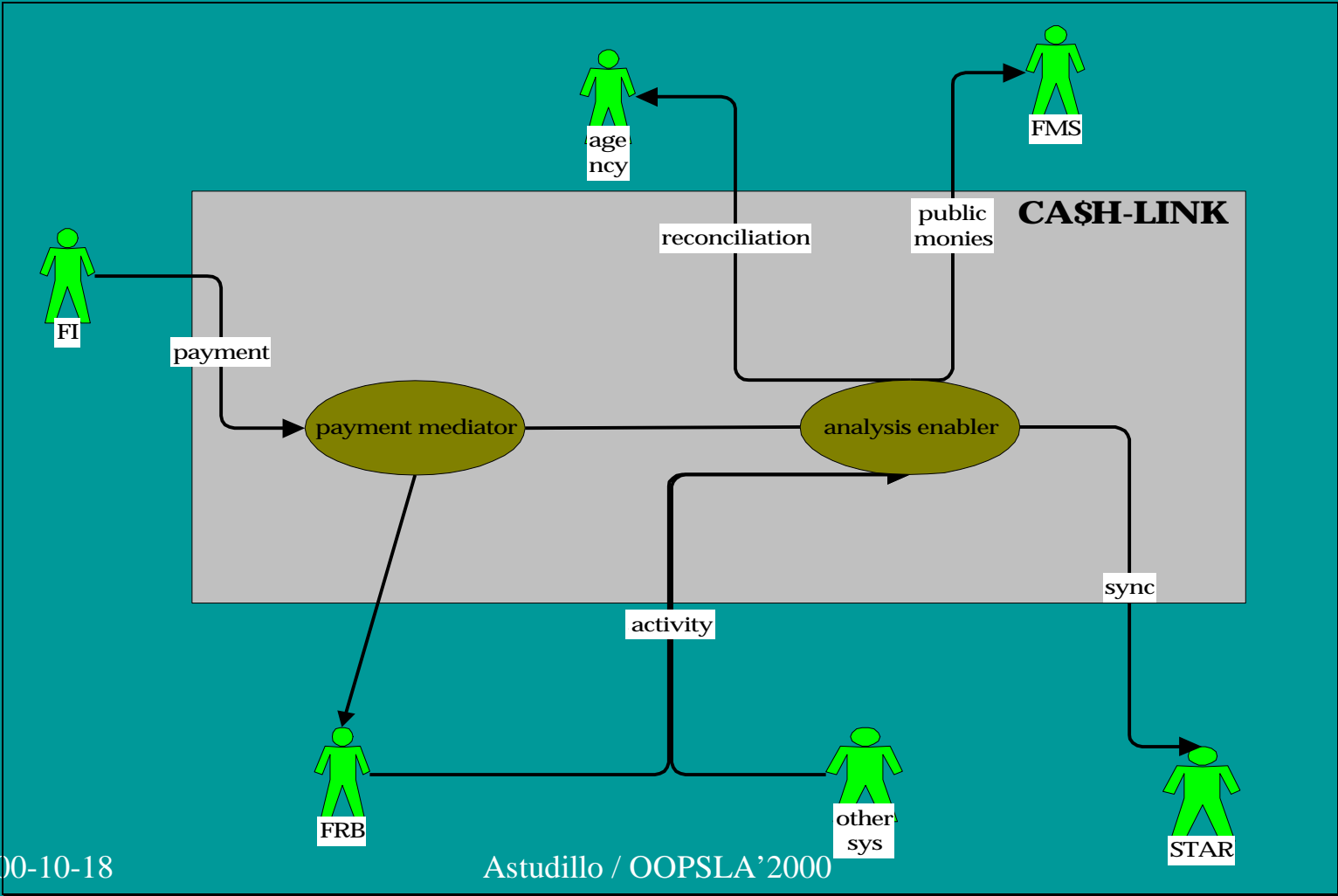
- 3) Services and connectors
- 4) Modules
 - pieces that can be treated as a single entity by a programming environment
- 5) Technical infrastructure
 - network/node topologies
 - deployment strategies

-
-
-

Typical components

- Typical component kinds:
 - lifecycle
 - allow CRED of a system-tracked entity
 - ... which can be internal or image of an external entity
 - ... providing basic retrieval strategies
 - reporting
 - generate reports on a system-tracked entity
 - ... providing complex/multiple retrieval strategies
 - ... with a content- & format-description mechanism
 - importing
 - accepts and converts external data meaningfully
 - conversion description mechanism

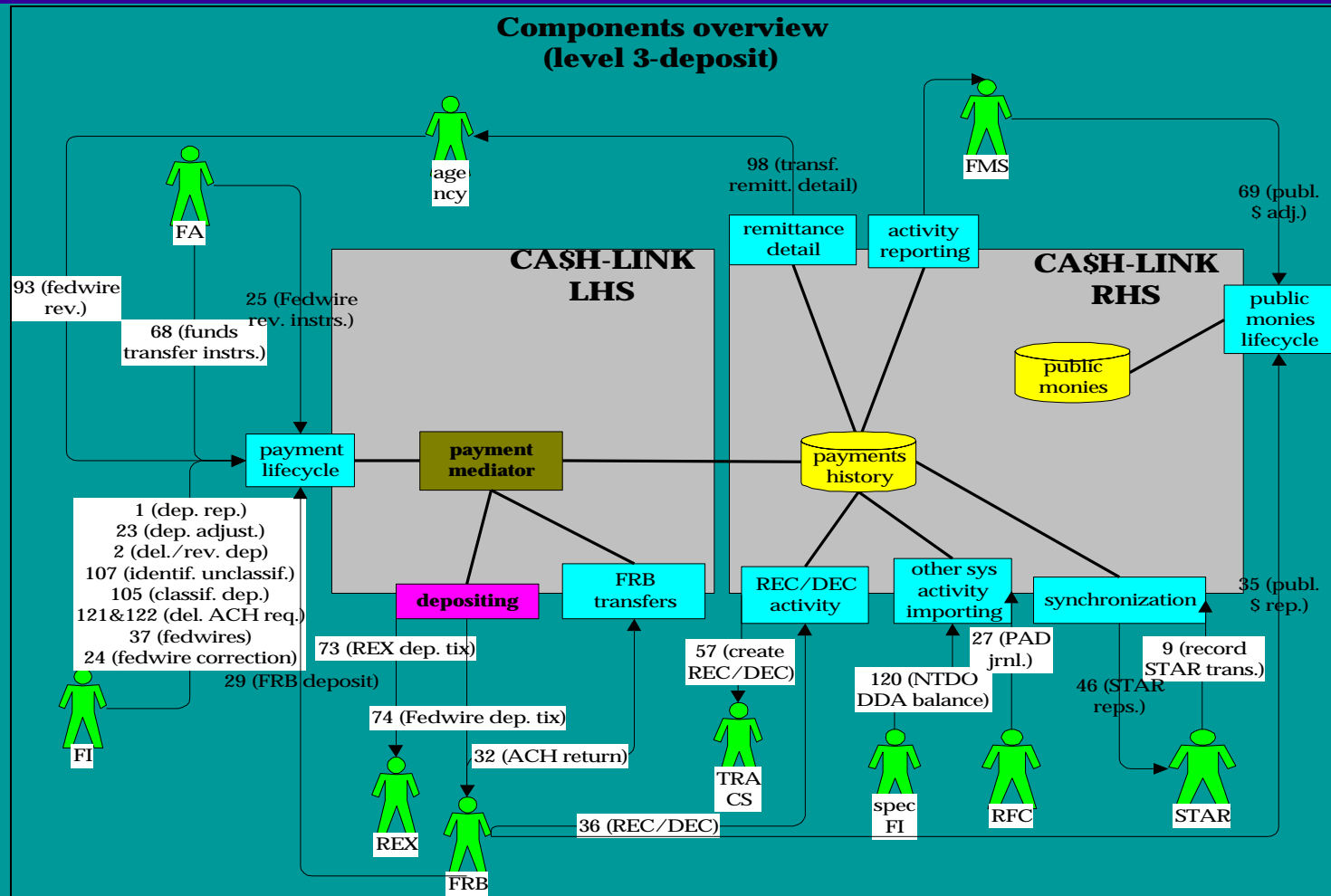
Level 1: lobes



2000-10-18

Astudillo / OOPSLA'2000

Level 2: components (deposit)



Level 2: traceability matrix

Event	Actors	Component(s)
#93 (Fedwire reversal)	Agency	Payment lifecycle
#68 (funds transfer instructions)	FA	Payment lifecycle
#25 (Fedwire reversal instructions)	FA	Payment lifecycle
#1 (deposit report)	FI	Payment lifecycle
#22 (deposit adjustment)	FI	Payment lifecycle
#3 (deposit deletion/reversal)	FI	Payment lifecycle
#102 (identif. Unclassified deposit)	FI	Payment lifecycle

Level 3: components refinement

Element	Element kind	Decomposition (used components)	Additional properties	Platform or product
Payment lifecycle			Load-balanced	
	Component	? Lifecycle	Fault-tolerant	Ad-hoc
Payment mediator	Interaction to FI/FA	? Collection	Fault-tolerant (duplicated to RHS)	
			Fault-tolerant (duplicated)	
Depositing	Component	? Persistence-LHS		Java/Oracle
			Legacy-based	
	Component	? Collection		

-
-
-
-
-
-
-
-
-
-

Conclusion



-
-
-
-
-
-
-
-

-
-
-

Conclusion

- Architectural experience may lead to identify business processes
 - lobes correspond to “real” things
 - business processes
- Multi-layered derivation is a good strategy
 - facilitates reasoning
 - provides traceability

-
-
-
-
-
-
-
-
-
-

How conceptual system architecture leads to business processes

Hernan Astudillo

Soluciones S.A.

hernan@acm.org

www.ime.usp.br/~ha/slides/oopsla2k.ppt

-
-
-
-
-
-
-
-