Service Oriented Architecture (SOA) Initiative: Kickoff Forum

Yes, You Can Buy That Valid Chart Strings

Brian Peters
Jason Schoober
Use Case

• University Services provides over two dozen services to the NU community
  – Mail
  – Motor pool
  – WildCARD
  – Travel
  – Parking
  – Moving
  – Lab services
  – Shuttles/charters
  – Copiers
  – Vending
  – Fitness/recreation
  – Stockrooms
  – Lab laundry
  – Equipment calibration
  – ...and more
Use Case

- These services can be requested online, via phone or email, in person, or on paper
  - Varying application sitting behind each function
- NUFinancials requires a valid chart string
  - Fund – 141 values
  - Department – 3,509 values
  - Project – 20,096 values

Almost 10 billion permutations
...but they’re not all valid
...is this one? Just trust me?
What Could Go Wrong?

• In most cases, there is no real-time validation that a chart string is valid
• Transactions are compiled for billing periodically
• Invalid chart strings are discovered
  – Combination of Fund-Department-Project doesn’t exist
  – Combination does not have a budget for this FY
  – Combination does not have sufficient funds
  – Project is a grant that has been CLOSED
The Workload of Invalid Chart Strings

- Determine the root cause of the invalid chart string
- Find somebody from whom to get a valid chart string
- If not fixed prior to billing, post charges anyway and then do transfers
- Not productive...
Somebody should validate these things...

• Chart string are validated today
  – Several interfaces from NUFinancials directly to other major systems (FASIS, SES, FAMIS, CATracks, etc.)
  – Daily file of valid chart strings made available via means of differing technical sophistication
    • XML file for import
    • Excel/CSV file for import
    • Cognos output to download

• But this assumes upkeep on the consumer’s part...
Chart String Validation using SOA

• Multiple services may make sense to meet the varying needs of different applications:
  – Valid chart string value – transaction
  – Valid chart string value file
  – Tentative budget check – transaction
  – Valid ChartField value – transaction
  – Valid ChartField value file
Valid chart string value – transaction

• A consuming system would inquire as to whether a Fund-Department-Project combination is valid
  – Has a budget in the current fiscal year (could be $0)
• In this mode, the inquiry would expect a pass/fail or yes/no answer in the course of validation in the consuming application
  – Ideally at the time that a transaction is first entered/captured in real time
Valid chart string value file

- Similar inquiry by consuming application vs. universe of valid Fund-Department-Project combinations
  - In this mode, the inquiry would likely query against a data set with either a pass/fail answer for each queried combination or a list of invalid combinations
  - A use case for this mode would include a need to validate a set of monthly charges in a recharge system prior to passing them to NUFinancials via a journal entry upload process
    - The journal entry upload could be its own web service!
Tentative budget check – transaction

• In most cases, strict budgetary control is not enforced in NUFinancials
  – Feinberg chart strings, some grants are the exceptions

• PeopleSoft Financials delivers a function to perform a “what if” budget check
  – Same as “Valid chart string value – transaction” use case, but would return a result of Valid, Warning, Error
  – If called in real-time, could prevent budget error before it happens (e.g., when entering funding in FASIS)
Valid ChartField value – transaction

• Systems may also wish to consume information about the values of a single ChartField
  – Department: ID, description, manager, admin contact, default fund
  – Project: ID, description, manager, admin contact, default fund, default department, purpose, start date, end date
  – Account: ID, description, account type
  – Fund: ID, description, fund category
Valid ChartField value – transaction

• In real-time, an application might inquire about attributes of a ChartField
• For example, for a Project ID
  – To determine the title of the project id
  – To understand the intended purpose of the project
  – To locate an individual to contact for approval prior the use of the project (or verify that the requester is a known approved used), etc.
Valid ChartField value file

• Much the same as above, a system may wish/need to consume the information about each ChartField in a periodic fashion
  – Determining whom to direct inquiries to if transactions are found to be invalid when preparing to send a monthly recharge file to NUFinancials
  – Other units may find this data valuable for other reasons, such as an input to a report or a simple mail merge task
Opportunities for re-use

• We all use the same chart of accounts
• Opportunity to be more timely, more precise in our validation
  – Get expenses posted to the right place the first time!
• Several unique interfaces could be retired if moved to published services via enterprise architecture