IMS API's

Dusty Rivers
Director z-Systems GT Software
drivers@gtsoftware.com

API !!!

Application Programming Interface (API) is a set of subroutine definitions, protocols, and tools for building application software.

In general terms, it is a set of clearly defined methods of communication between various software components.

A good API makes it easier to develop a computer program by providing all the building blocks, which are then put together by the programmer.

An API may be for a web-based system, operating system, database system, computer hardware or software library.

An API specification can take many forms, but often includes specifications for routines, data structures, object classes, variables or remote calls. POSIX, Microsoft Windows API, the C++ Standard Template Library and Java APIs are examples of different forms of APIs.

Documentation for the API is usually provided to facilitate usage.

Wikipedia….

Are you kidding me???? IMS!

IMS


COBOL
ASM
PL/1
RACF
3270
Other!
IMS/TM
Transactions
CICS
Transactions
DB2
VSAM
DL/1

Another Bridge!!

API's

REST/JSON,SOAP,JDBC,ODBC

GTSoftware®

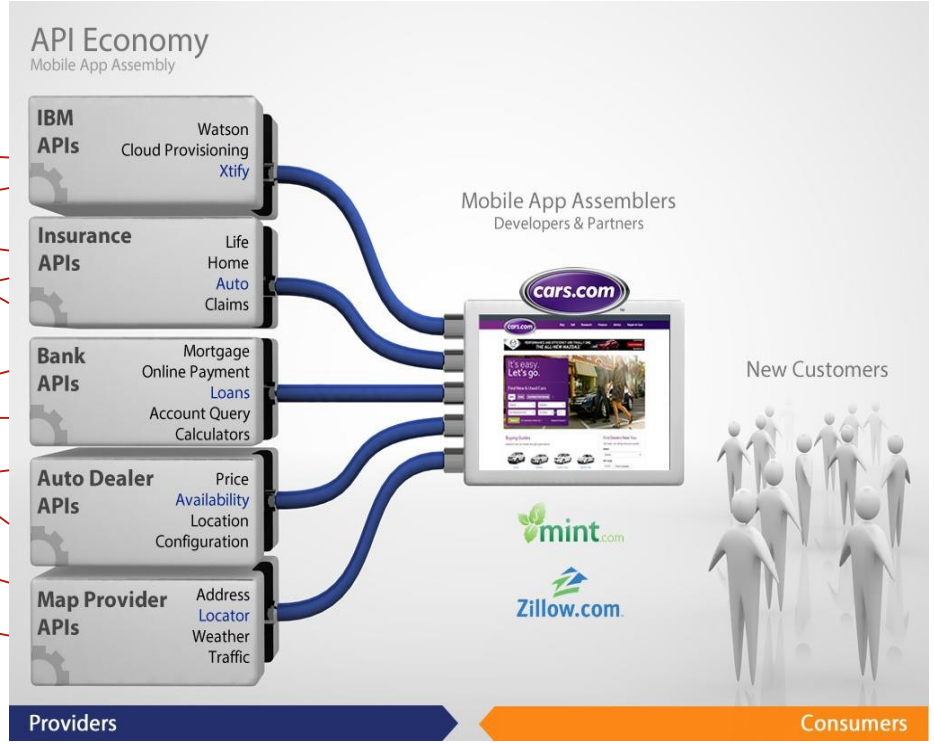
Atomic APIs

API's

REST/JSON,SOAP,JDBC,ODBC

GetSavng{}
GetChkng{}
GetCredit{}
GetMortg{}

GetPart{}
GetDesc{}
GetInv{}
GetOrder{}



GTSoftware®

API's

REST/JSON,SOAP,JDBC,ODBC

Atomic APIs

GetSavng{}
GetChkng{}
GetCredit{}
GetMortg{}

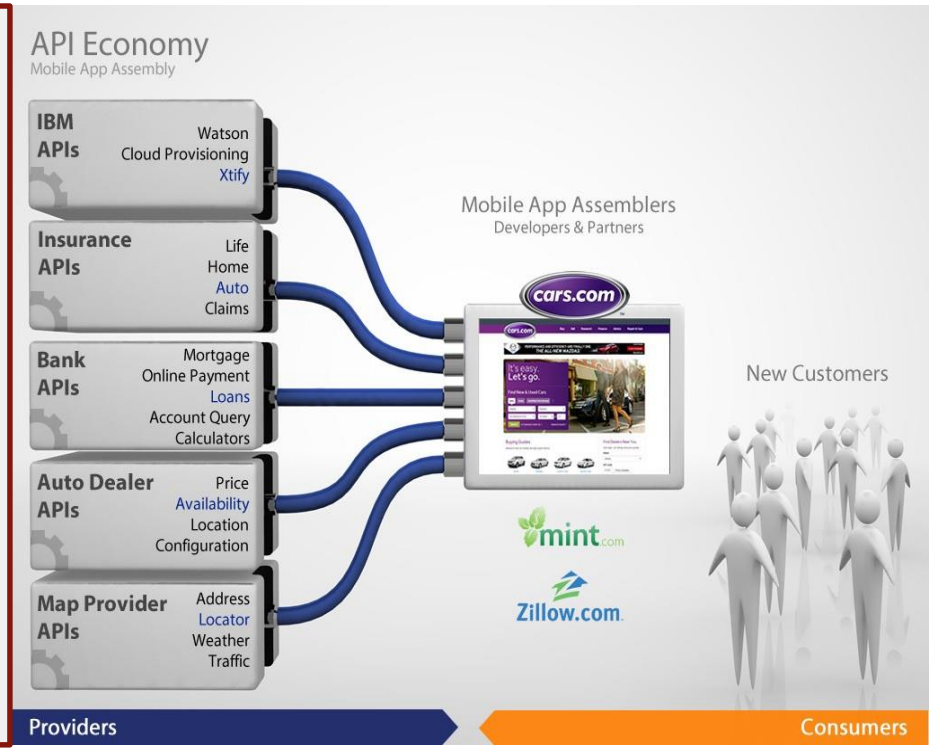
GetPart{}
GetDesc{}
GetInv{}
GetOrder{}

Composite APIs

GetCust{}

GetStatus{}

ListCust{}



- COBOL and PL/1
- IMS Transactions
- IMS Transactions Combined
- IMS Conversational
- IMS Multi-Segment Output

COBOL and PL/1

GTSoftware®

THE GOOD	THE BAD	THE UGLY
All Data Structures Supported	Some structures don't map well to distributed Apps	Comp-3, Binary , ODO REDEFINES
All can be exposed as service inputs/outputs	Names in COBOL-PL/1 may be cryptic and need to be renamed	c3-7-proj1-open-partnum12 = partNUM
Can expose existing programs without changes	May need more data to drive than the app knows	Message switches, and other calls

IMS Transactions

GTSoftware®

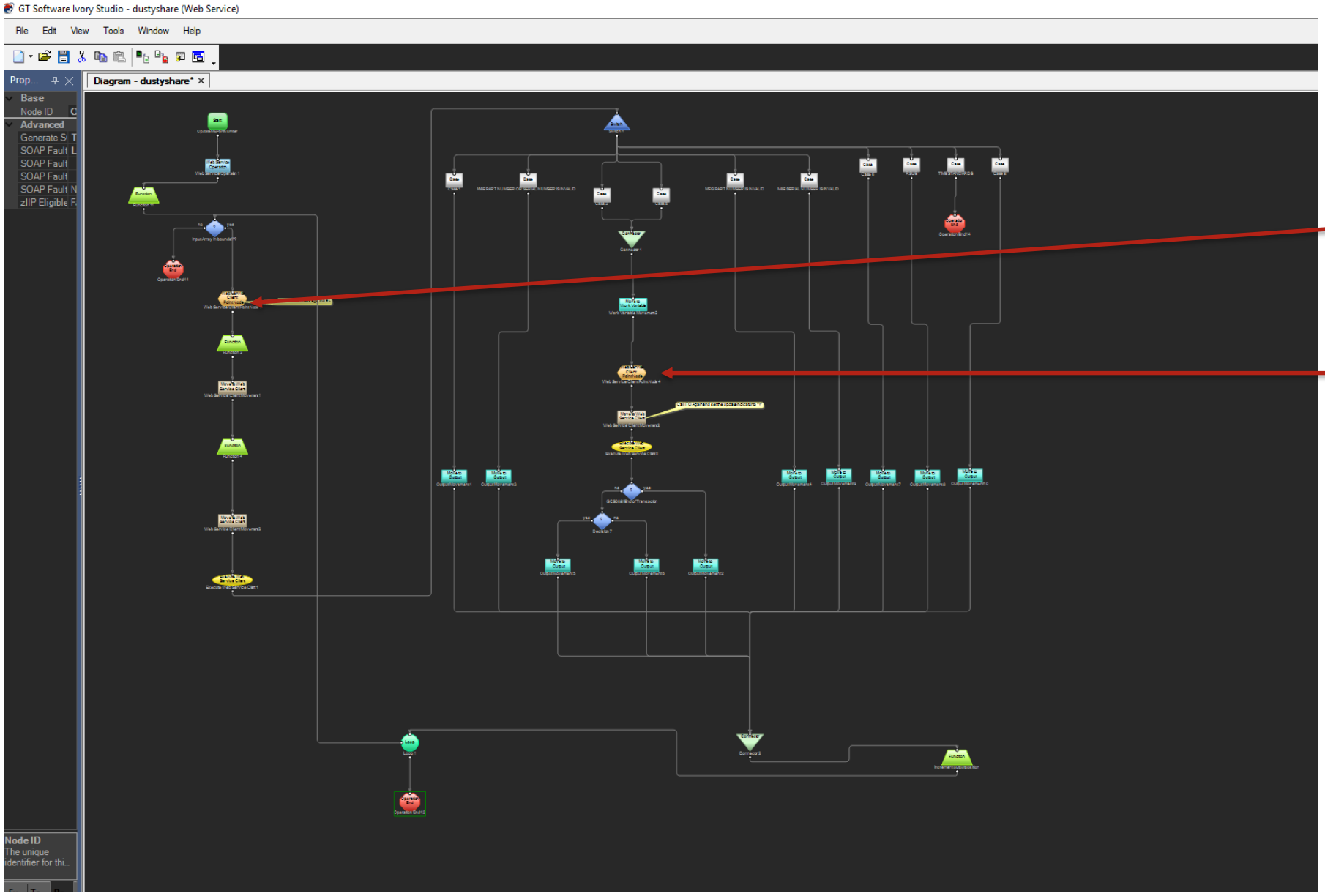
THE GOOD	THE BAD	THE UGLY
Existing Transactions can be exposed as REST or SOAP	A Transaction may be too fine grained	Multiple Transactions may have to be used in service
Data from transaction returned as a service output	Data me be to convoluted to use in service	Volume of data may be too large to return to distributed client
PFKEY = TRANCODE	Maybe need multiple Trans	

IMS Transactions Combined

GTSoftware®

THE GOOD	THE BAD	THE UGLY
Combine Transactions in one service	May not work well with others	API's that run for minutes
Use Conversational Transactions	Long running conversations may be long running API's	No understanding of conversational impact
No Code re-write	May be easier to combine logic to keep from calling multiples transactions	

Multiple Transactions (good/bad/ugly!)



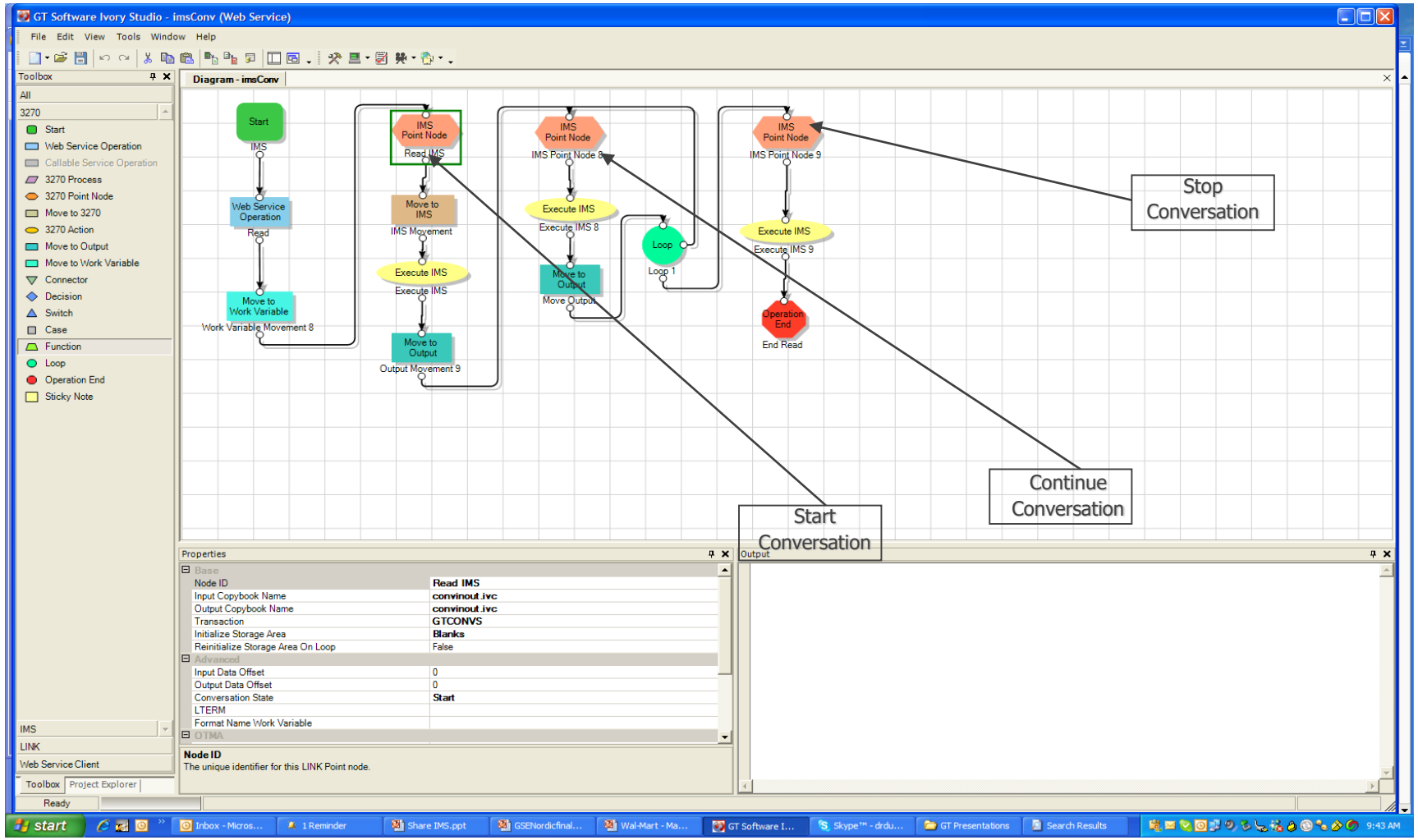
Transaction 1 (list)

Transaction 2 (process)

IMS Conversational

THE GOOD	THE BAD	THE UGLY
Wrap a conversation in a service	Wrap a conversation in a service	Wrap a conversation in a service
Use Conversational Transactions	Long running conversations may be long running API's	Conversational rollback

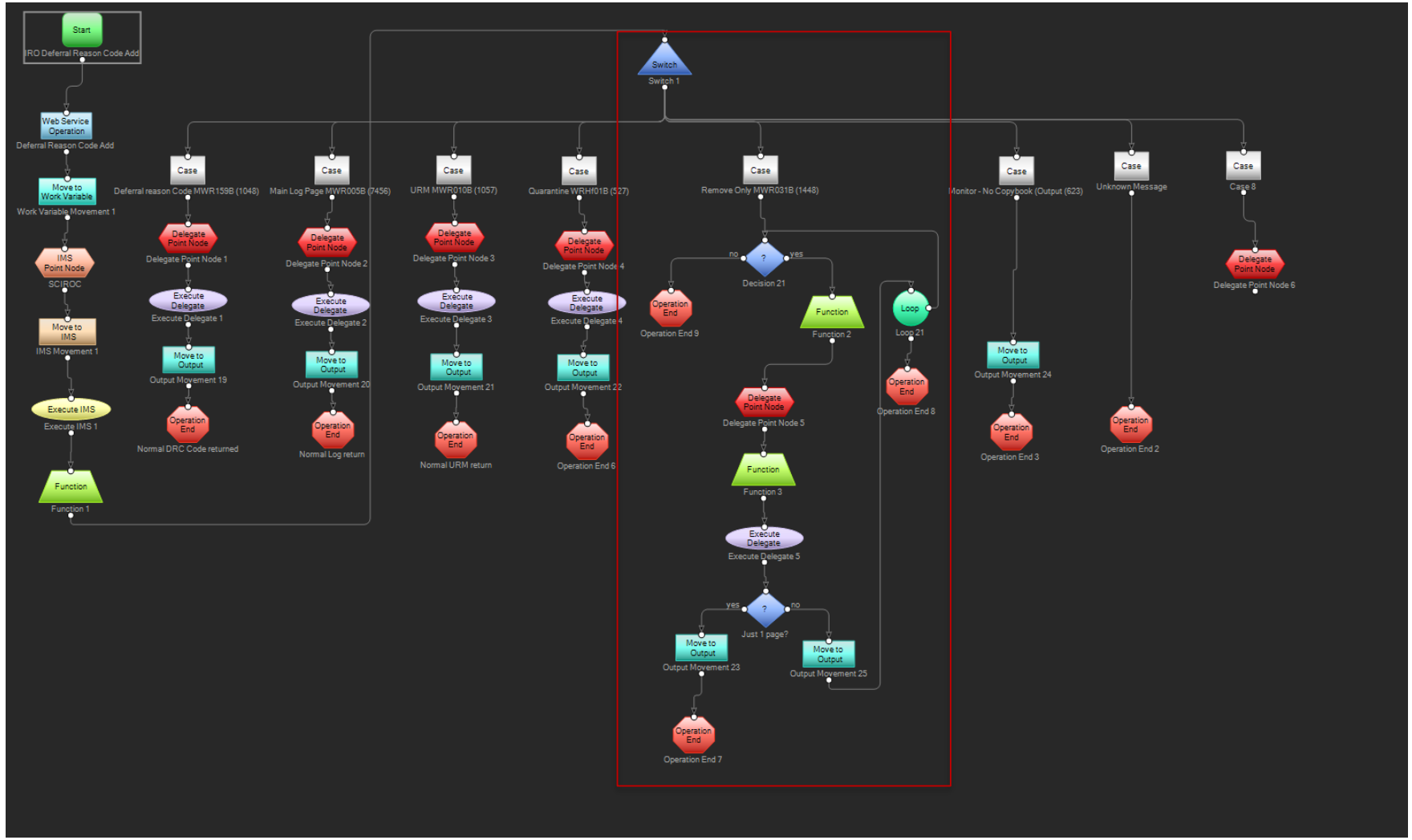
IMS Conversational Tran as a Service



IMS Multi-Segment Output

THE GOOD	THE BAD	THE UGLY
Multiple Segment output can be returned to client	May be variable Length in one response	May be variable length multi-segment response

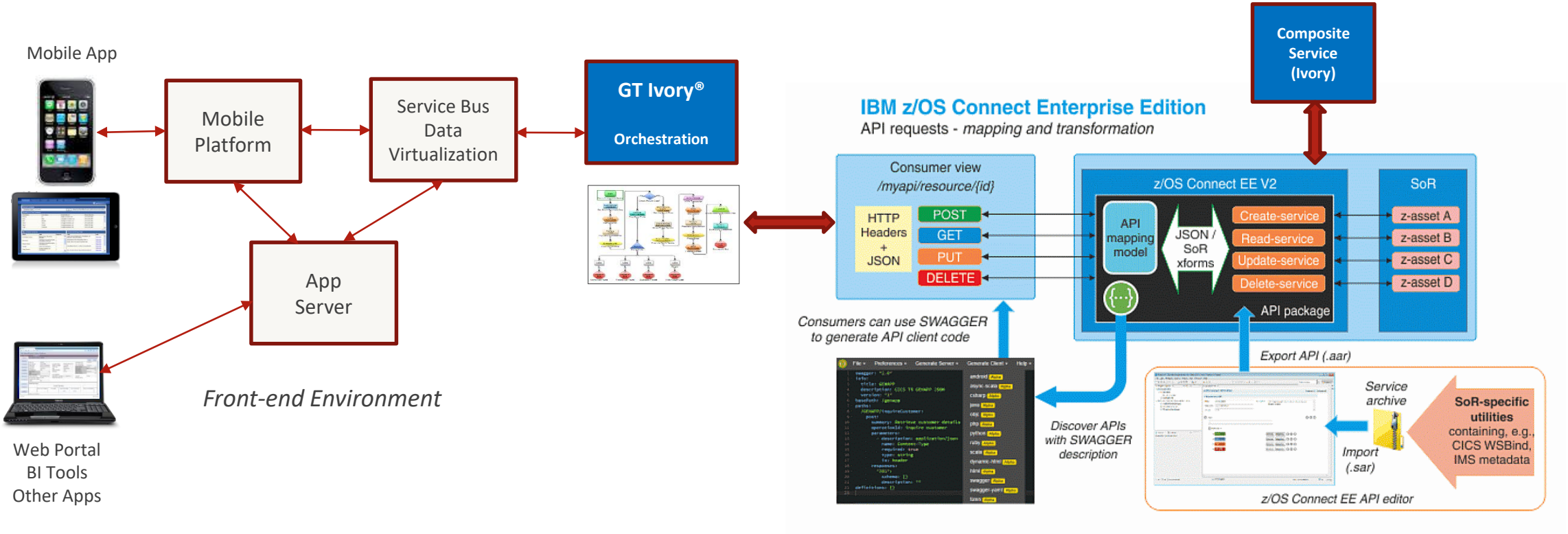
IMS Multi-Segment Output



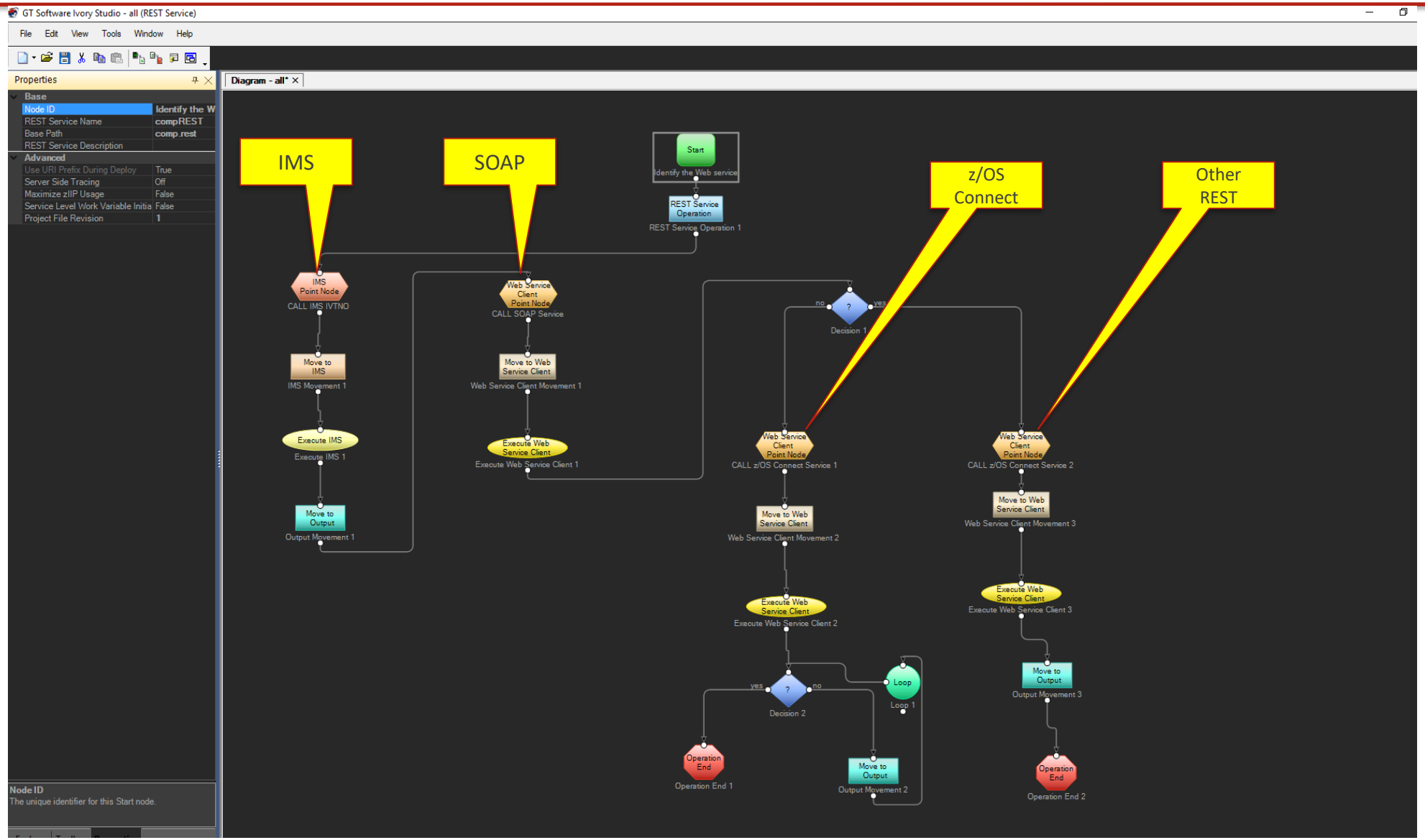
Examples

- z/OS Connect
- Data Virtualization
- Red Hat
- BI/BA

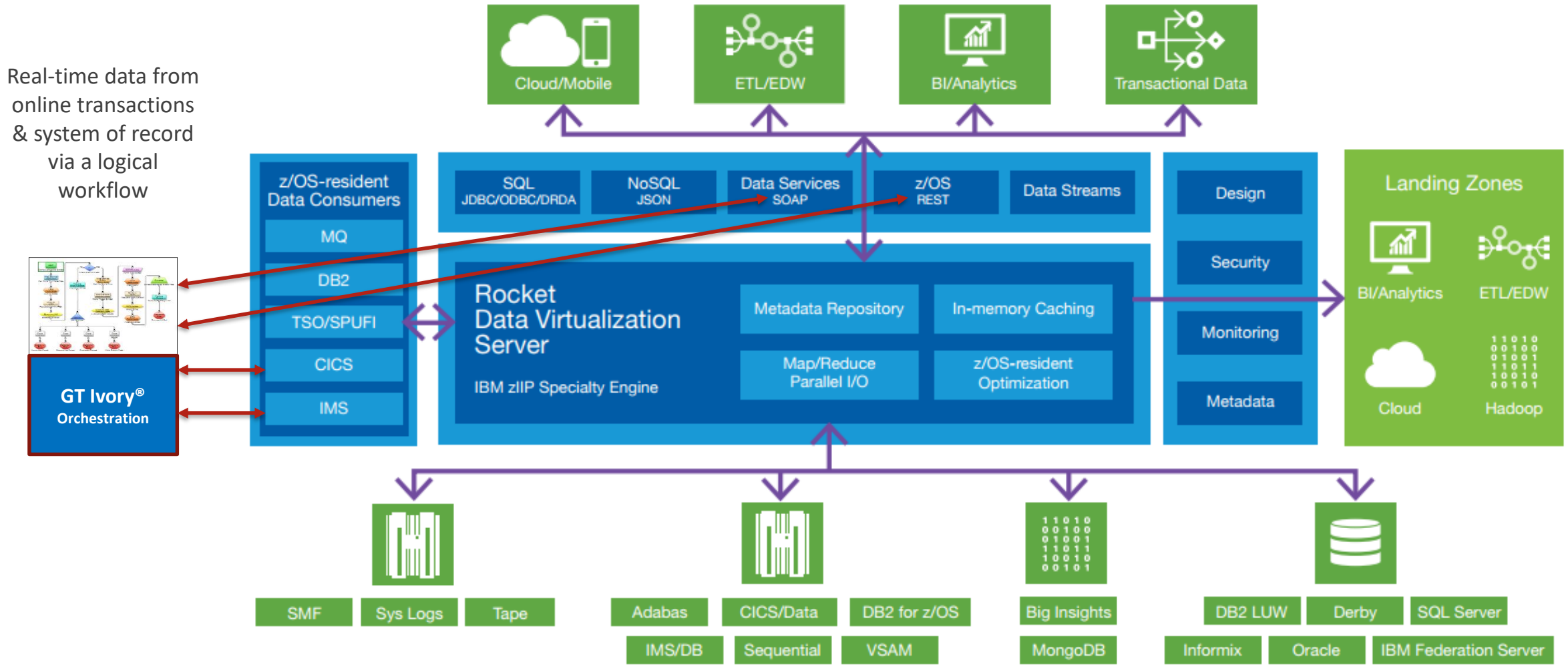
GT Ivory Orchestration with IBM z/OS Connect



Orchestration Example



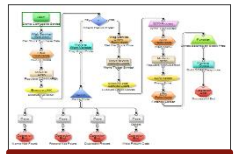
GT Ivory Orchestration with Data Virtualization



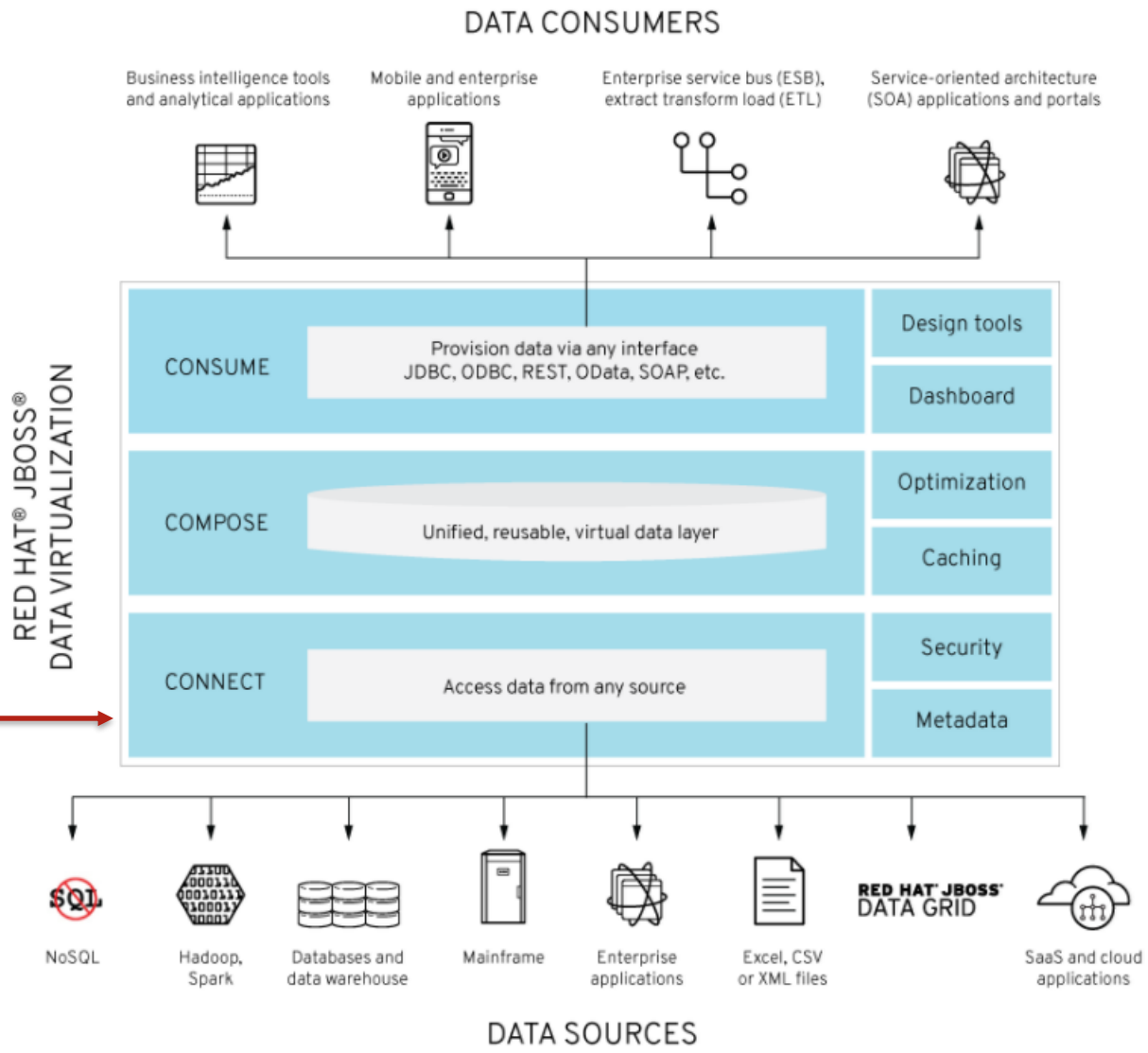
* Update processing better performed via online transaction program logic to keep databases and file in sync.

GT Ivory Orchestration with Data Virtualization

Real-time data from online transactions & system of record via a logical workflow



GT Ivory®
Orchestration



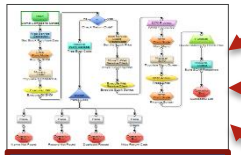
GT Ivory Orchestration with BI & Data Integration

GTSoftware®

Real-time Self-service Analytics

Real-time data from online transactions & system of record via a logical workflow

IBM Mainframe
(zOS or VSE)

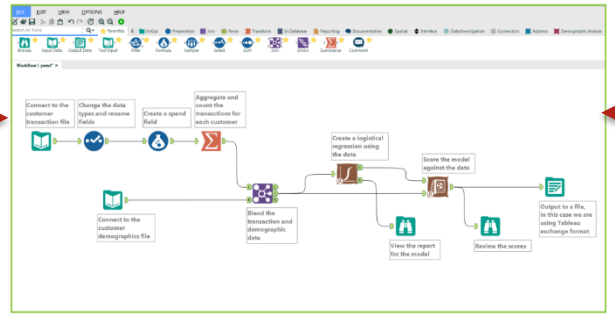


GT Ivory®
Orchestration

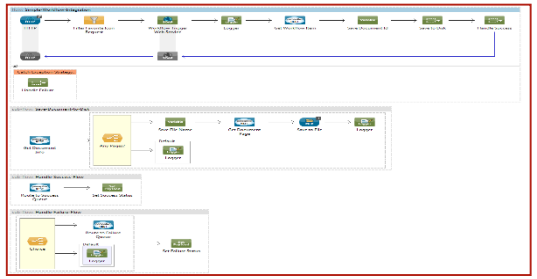
REST API

SOAP API

alteryx talend pentaho



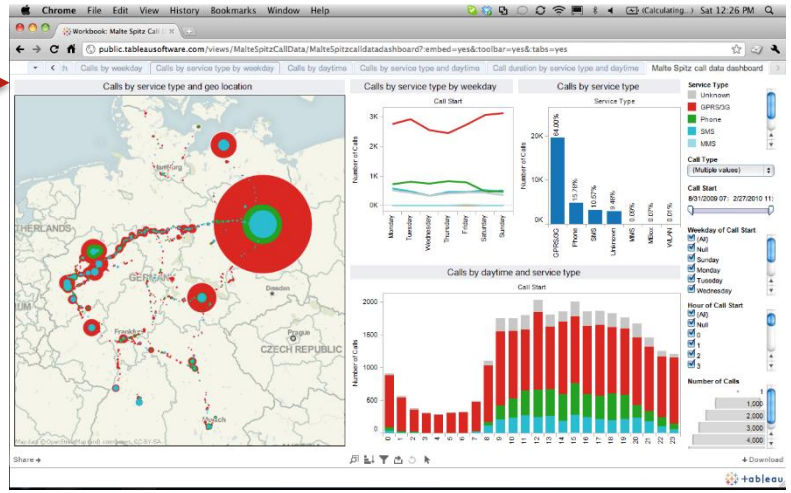
MuleSoft informatica TIBCO



DOMO Qlik

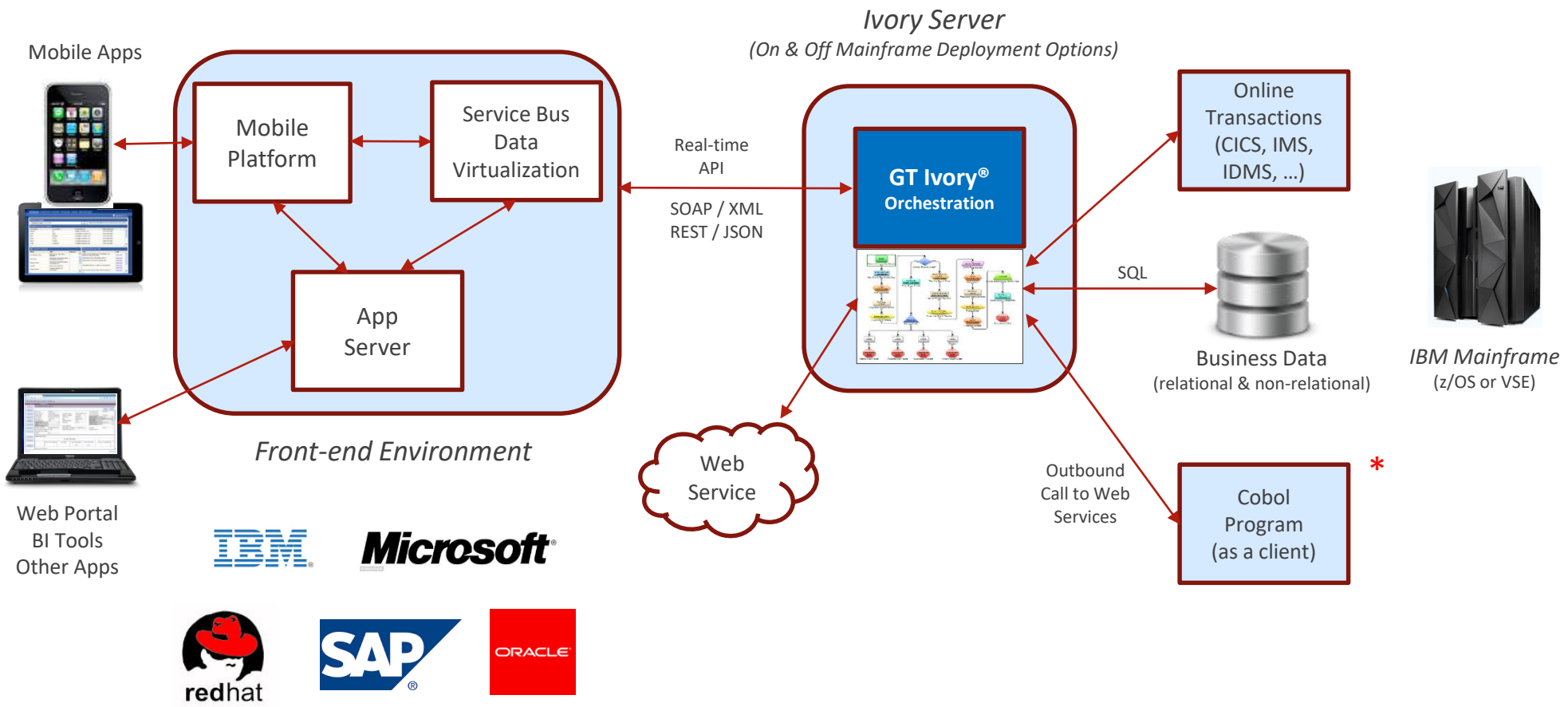
Power BI birst

tableau SOFTWARE



* Bad data "in" will result in bad information "out"

GT Ivory Orchestration for z/OS and VSE

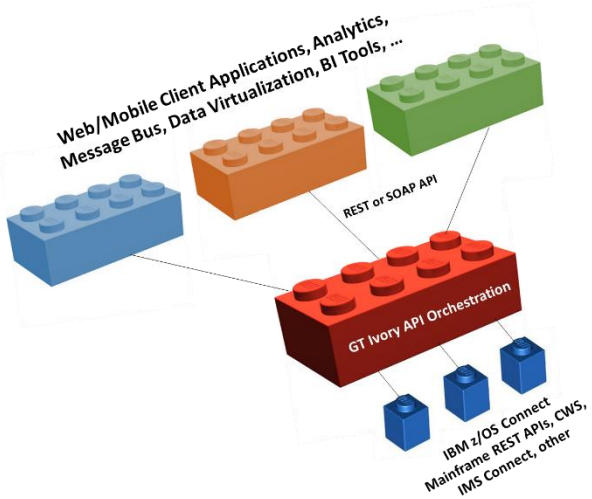


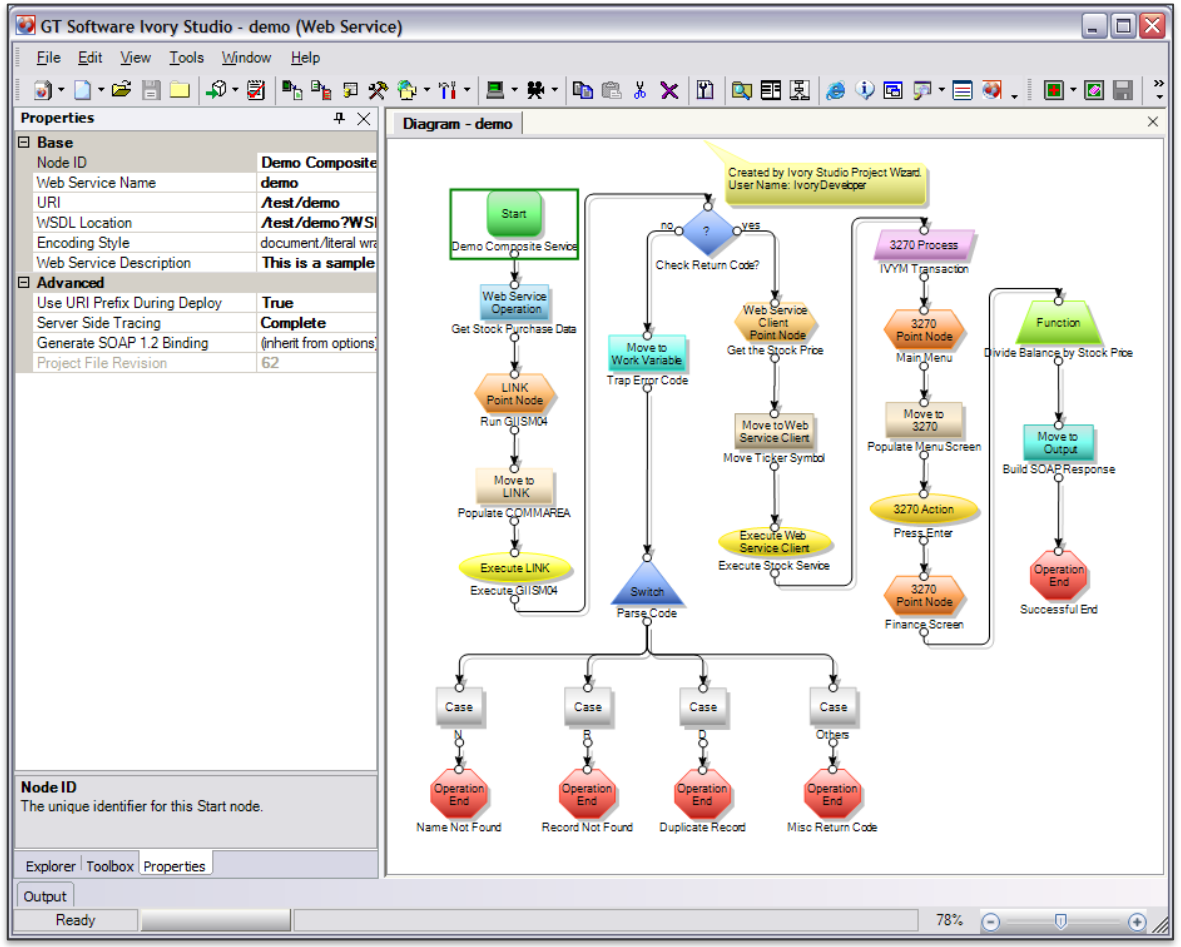
IBM Mainframe
(z/OS or VSE)

Connectors

- *CICS TRANSACTION GATEWAY*
- *CICS WEB SERVICES (CWS)*
- *IMS CONNECT*
- ***IMS SOAP GATEWAY*** *
- *Z/OS CONNECT*
- *HOSTBRIDGE®*
- *TN3270*

GT Ivory Orchestration Workflow



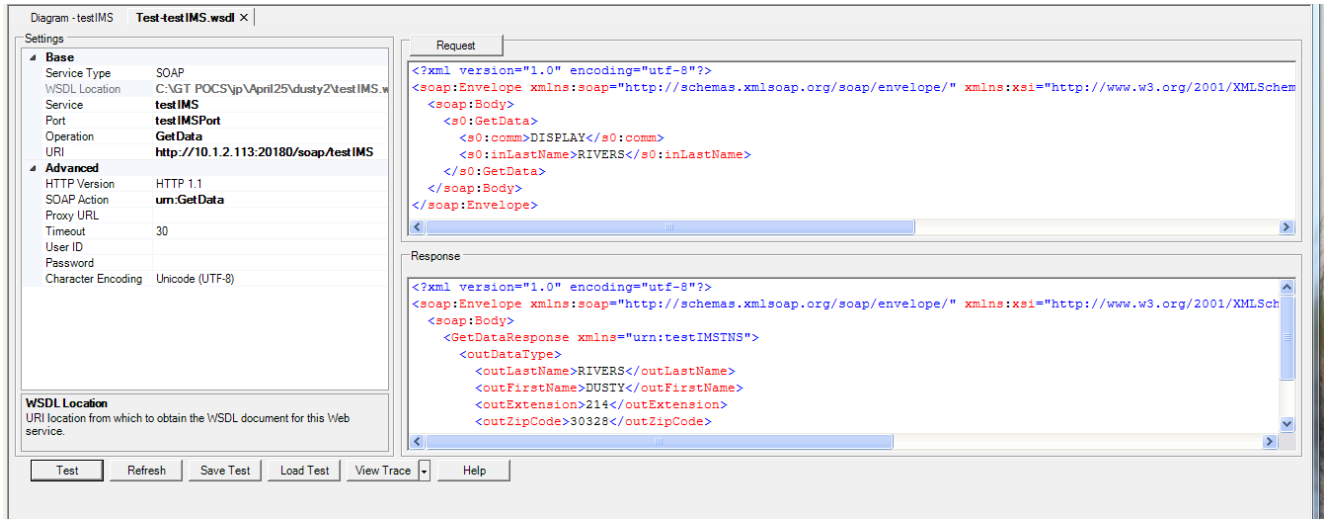


Intelligent Composite API:

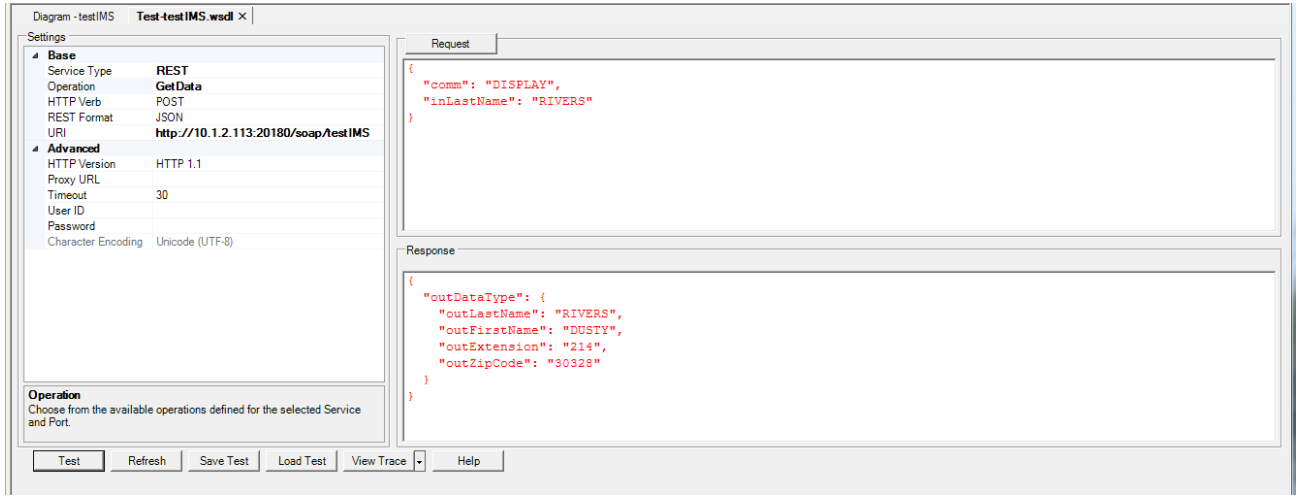
- Multiple transactions
- Multiple data sources
- External web services and APIs
- Conditional Logic
- Error handling
- Governance and security
- Drag-and-drop (no coding) SDK
- Shared ‘business’ APIs across consumers
- No ‘low level’ coding and management of mainframe connectors
- Easy, fast, and agile development

GT Ivory Generated SOAP and REST APIs

SOAP Service Example

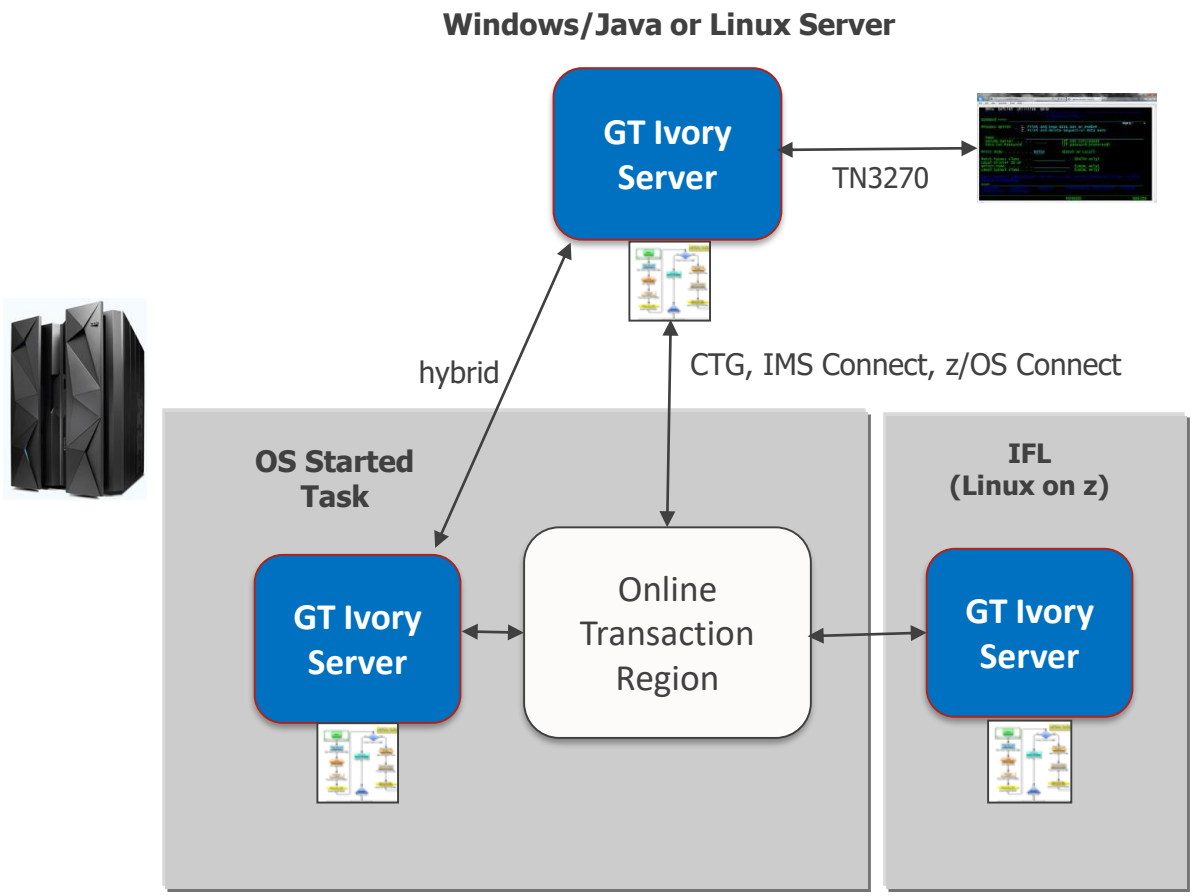


REST Service Example

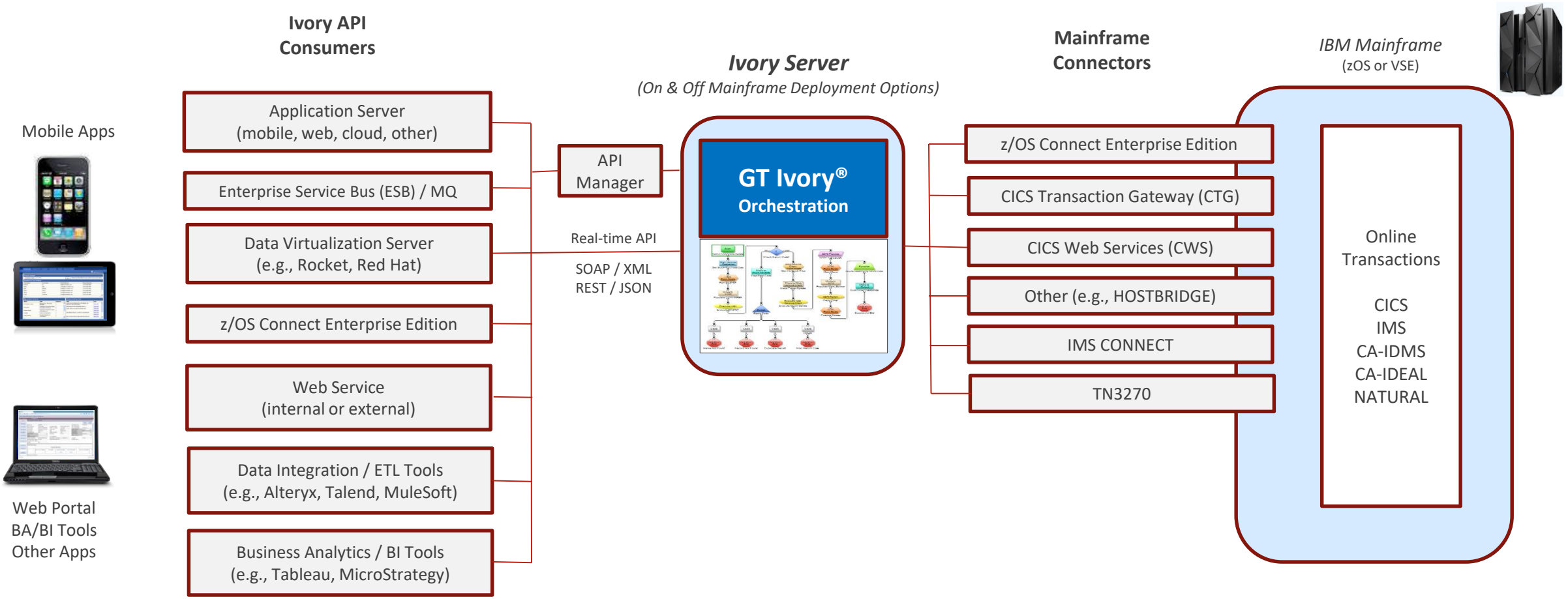


- Wizard within Ivory Studio generates the service definition from the orchestration workflow
- A service can be created as SOAP/XML or REST/JSON
- Can have an orchestration exposed as both a SOAP and REST service
- Services can be tested real-time with multiple levels of tracing for debugging
- A test (input data) can be saved and repeated in support of iterative development

GT Ivory On and Off Mainframe Deployment Options



GT Ivory Orchestration Uses



Transformers **Voltron**

- Founded in 1982 (HQ in Atlanta, GA)
- More than 30 years of market leadership
- Focused on real-time mainframe integration for strategic business initiatives
- Broad experience across all mainframe and distributed environments
- Worldwide cross-industry customers and strategic partnerships

www.GTSoftware.com