MICRO® FOCUS

Service Virtualization

Getting higher quality software to market in less time, for less cost

jiri.tejkl@microfocus.com

Oct 2019

Getting higher quality software to market in less time, for less cost

- Challenges of enterprise application delivery
- Service Virtualization concepts
- Micro Focus Service Virtualization introduction
- Success stories

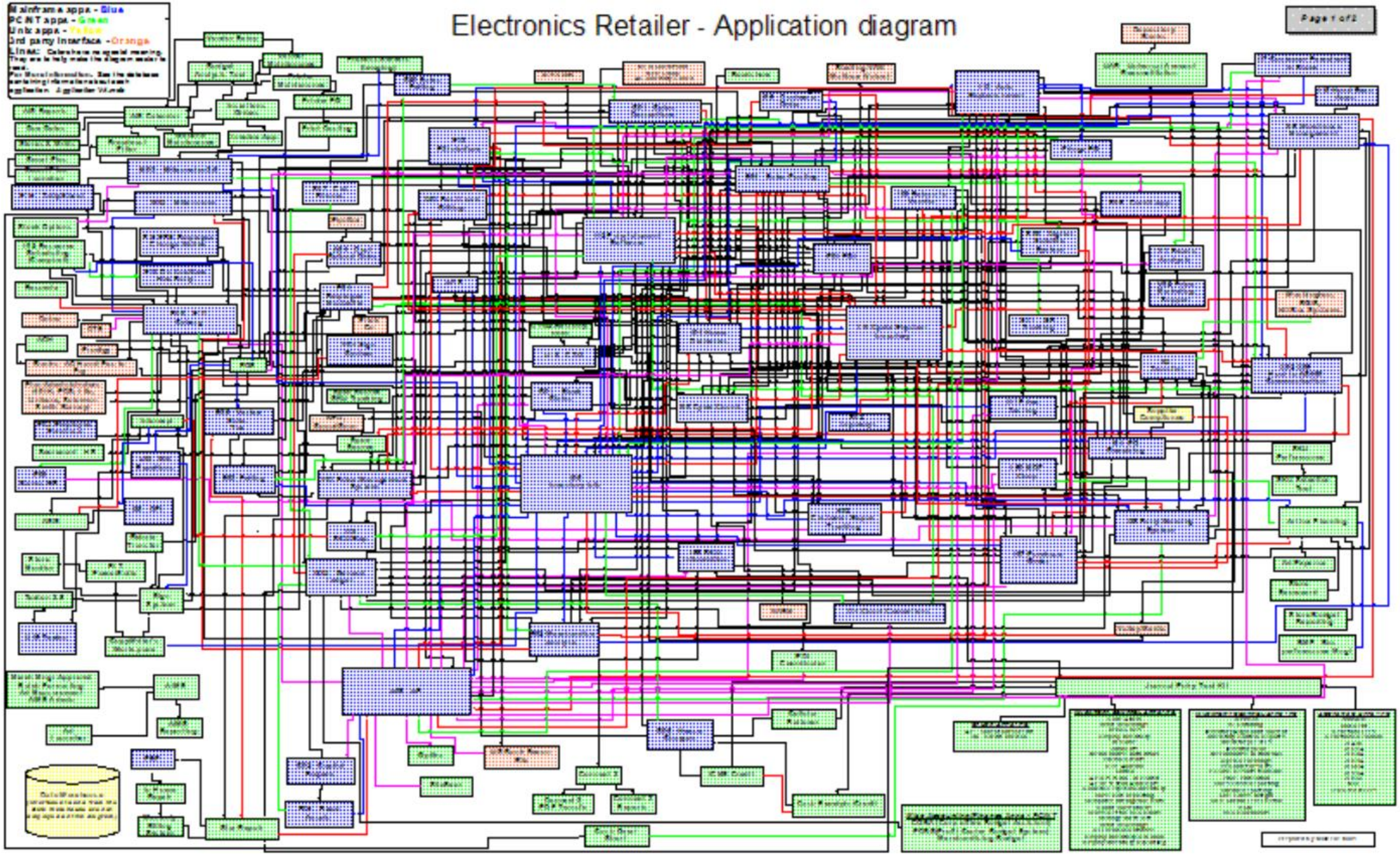
The Journey To Enterprise DevOps

Continuous Delivery maturity matrix

	Novice	Beginner	Intermediary	Advanced	Expert
Build	Verification before commit run in developer's Workspace Common nightly build	CI server builds on commit Artifacts are managed	No build scripts -only configurations Dependencies are managed	Distributed builds Staged build sequence	Build from VM CI server orchestrate VMs
Test + QA	Unit Test Code Coverage	Metrics on technical debt & compliance Mock-up's & proxies	Peer-reviews Automated Functional Test	Test Data Test in target	Automated Acceptance Test
SCM	"Early Branching" Branches used for releases Merges are rare	"Late branching" Branches used for work isolation Merges are common	Pre-tested Commits Integration branch is pristine	All commits are tied to tasks Individual history rewrites In DVCS	Release notes & traceability analysis are generated automatically
Visibility	Build status is notified to committer	Latest build status is available to all stakeholders	Trend reports Build status can be subscribed to (pull vs push)	Monitors in work areas show real-time status	Build reports and statistics are shared with customer and public

Chris Shayan, 2013

Electronics Retailer - Application diagram

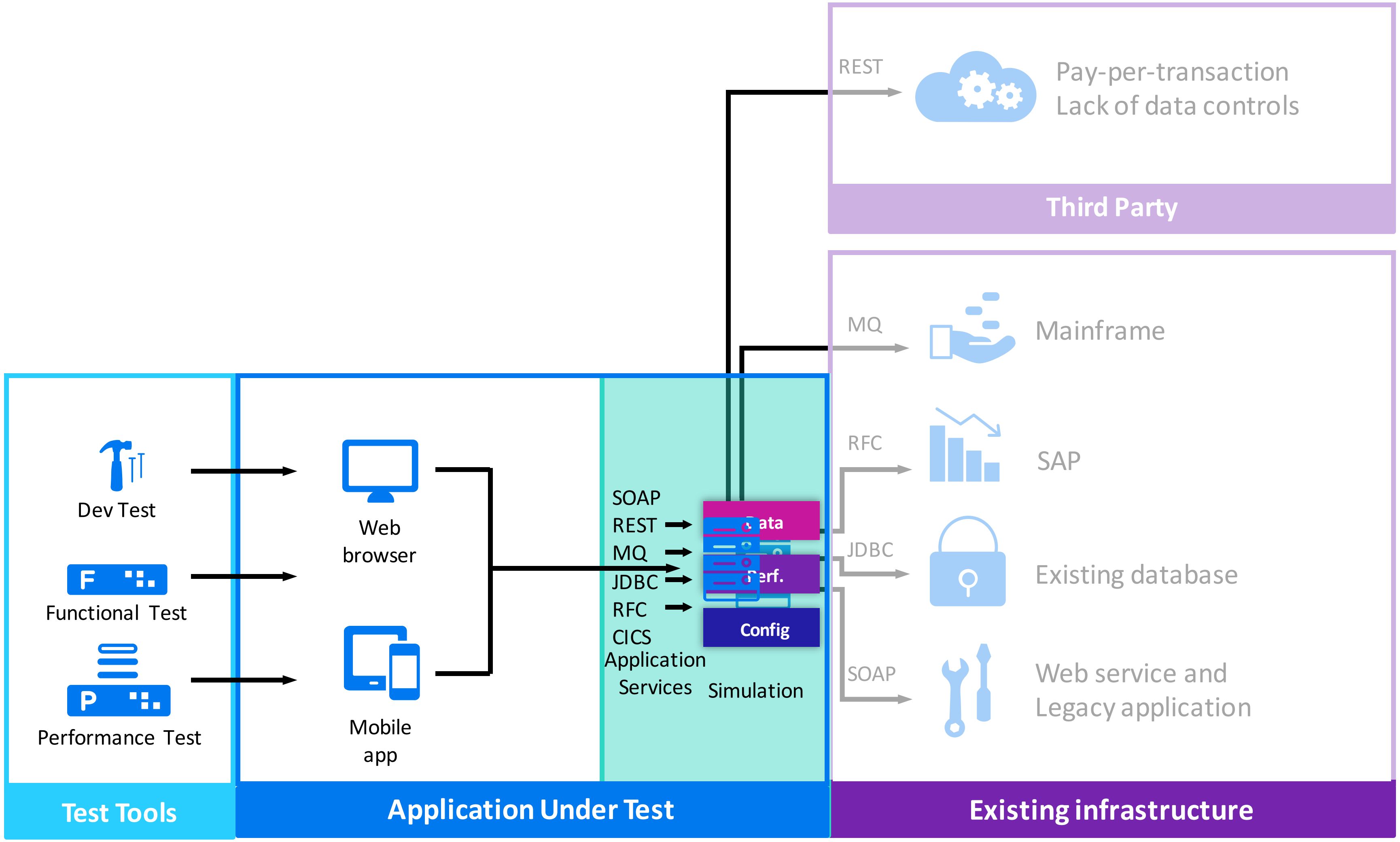

Mainframe apps - Blue
PC NT apps - Green
Unix apps - Yellow
3rd party interface - Orange
Lines: Colors have no special meaning. They are to help make the diagram easier to read.
For More information, See the database containing information about each application.
Page 1 of 2

Agile application delivery challenges

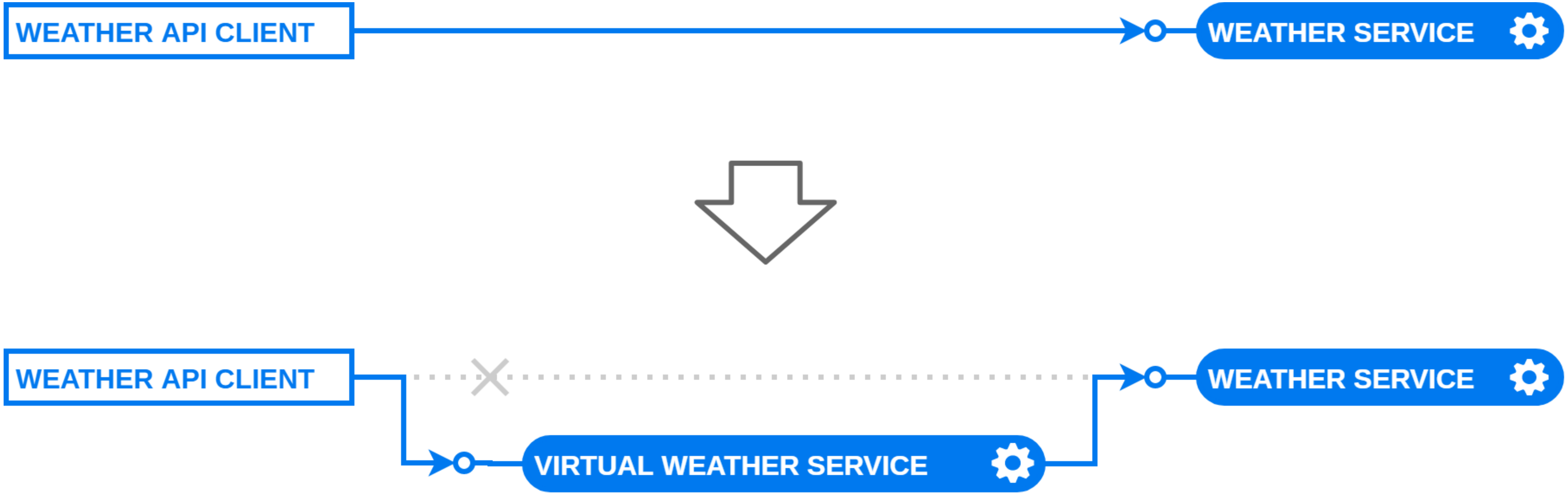
Slow turnaround

Service Virtualization Concepts

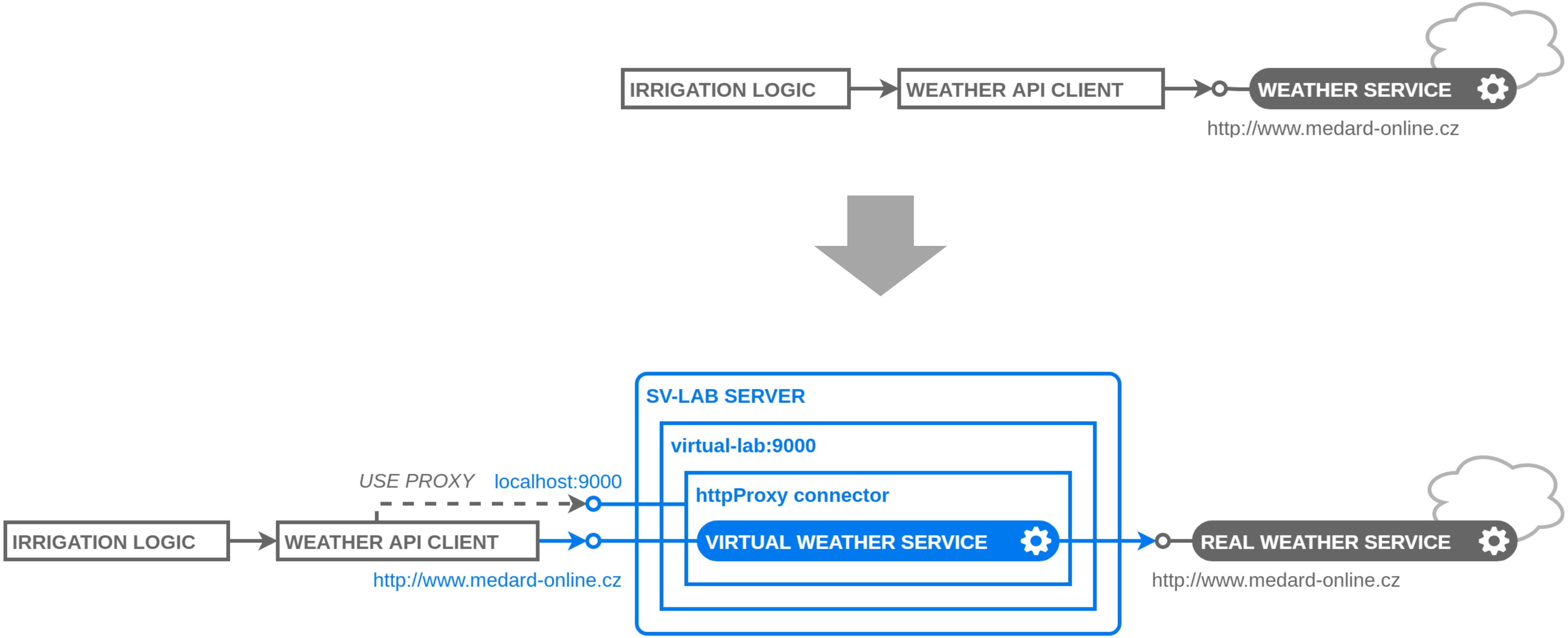
Service Virtualization concept



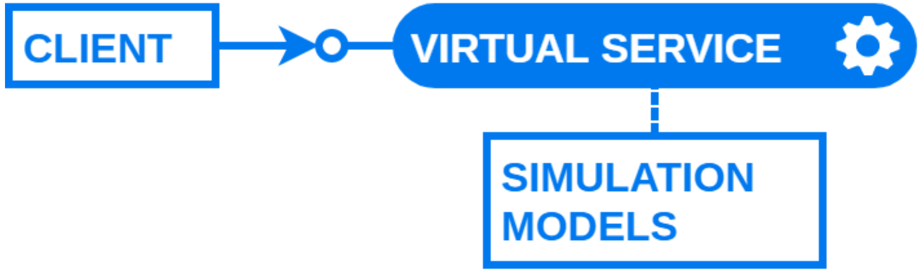
Virtual Service



Virtualization of REST Service



SIMULATION MODE:



MOCKS/STUBS/PROXIES

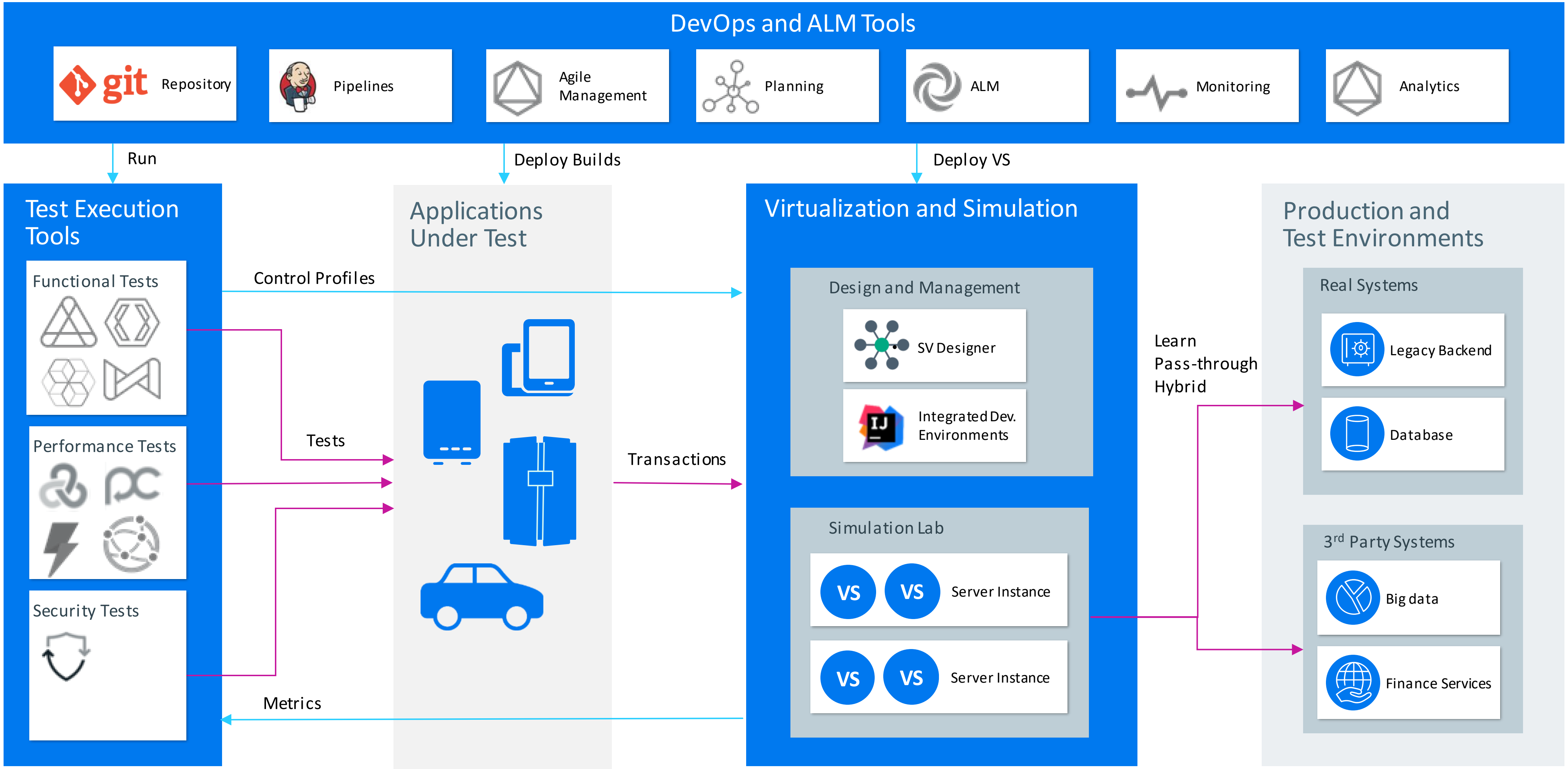
- Implemented in code

VIRTUAL SERVICES

- Rule/message-based declarative models

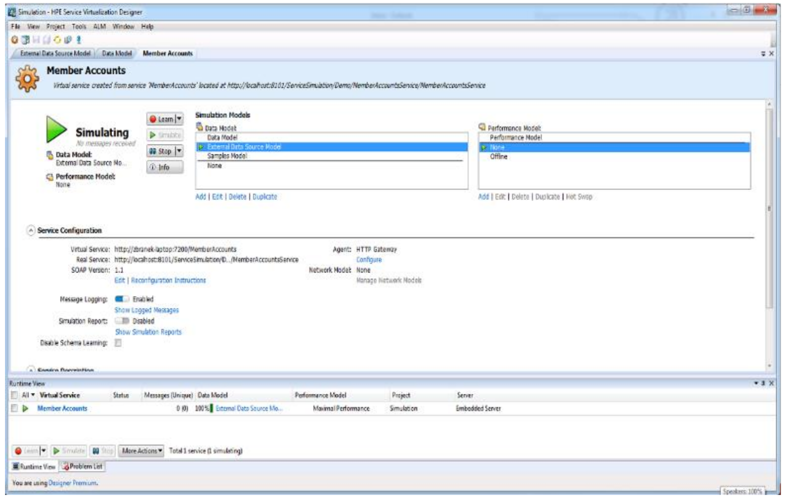
Micro Focus Service Virtualization Introduction

SV integrated to Micro Focus ADM toolset



DevOps and ALM Tools

Core components of Micro Focus Service Virtualization


Simulation - HPE Service Virtualization Designer
File View Project Tools ALM Window Help
External Data Source Model
Data Model
Member Accounts
Learn
Simulate
Stop
Info
Simulation Models
Add | Edit | Delete | Duplicate
Add | Edit | Delete | Duplicate | Hot Swap
Service Configuration
Service Description
Runtime View
Learn
Simulate
Stop
More Actions
Runtime View
Problem List
Speakers: 100%

Member Accounts

Virtual service created from service 'MemberAccounts' located at http://localhost:8101/ServiceSimulation/Demo/MemberAccountsService/MemberAccountsService

Simulating

No messages received

Data Model: External Data Source Mo...

Performance Model: None

Data Model:
- Data Model
- External Data Source Model
- Samples Model
- None

Performance Model:
- Performance Model
- None
- Offline

Virtual Service: http://zbranek-laptop:7200/MemberAccounts

Real Service: http://localhost:8101/ServiceSimulation/D.../MemberAccountsService

SOAP Version: 1.1

Edit | Reconfiguration Instructions

Agent: HTTP Gateway

Configure

Network Model: None

Manage Network Models

Message Logging: Enabled

Show Logged Messages

Simulation Report: Disabled

Show Simulation Reports

Disable Schema Learning: □

□ All	Virtual Service	Status	Messages (Unique)	Data Model	Performance Model	Project	Server
□	Member Accounts		0 (0)	100% External Data Source Mo...	Maximal Performance	Simulation	Embedded Server

Total 1 service (1 simulating)

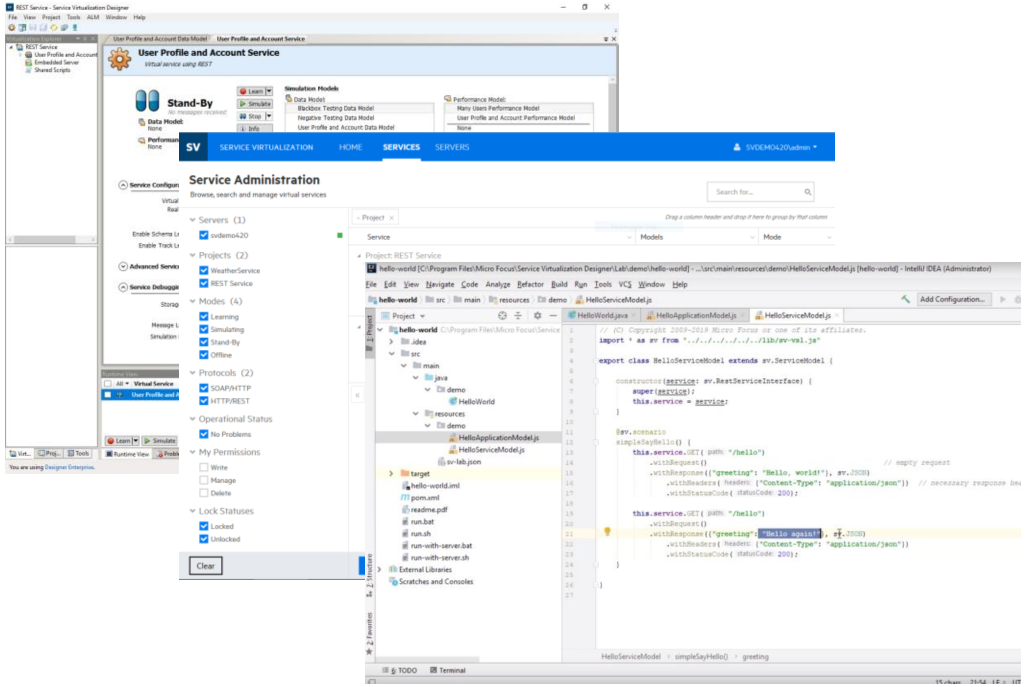
You are using Designer Premium.

SV Designer

- Visual modeling of Virtual Services
- Easy to use desktop IDE
- Embedded SV runtime
- Limited free version
- ALM/QC and SCM integrations

MF Service Virtualization key capabilities and characteristics

The industry’s most easy-to-use solution to design and run Virtual Services

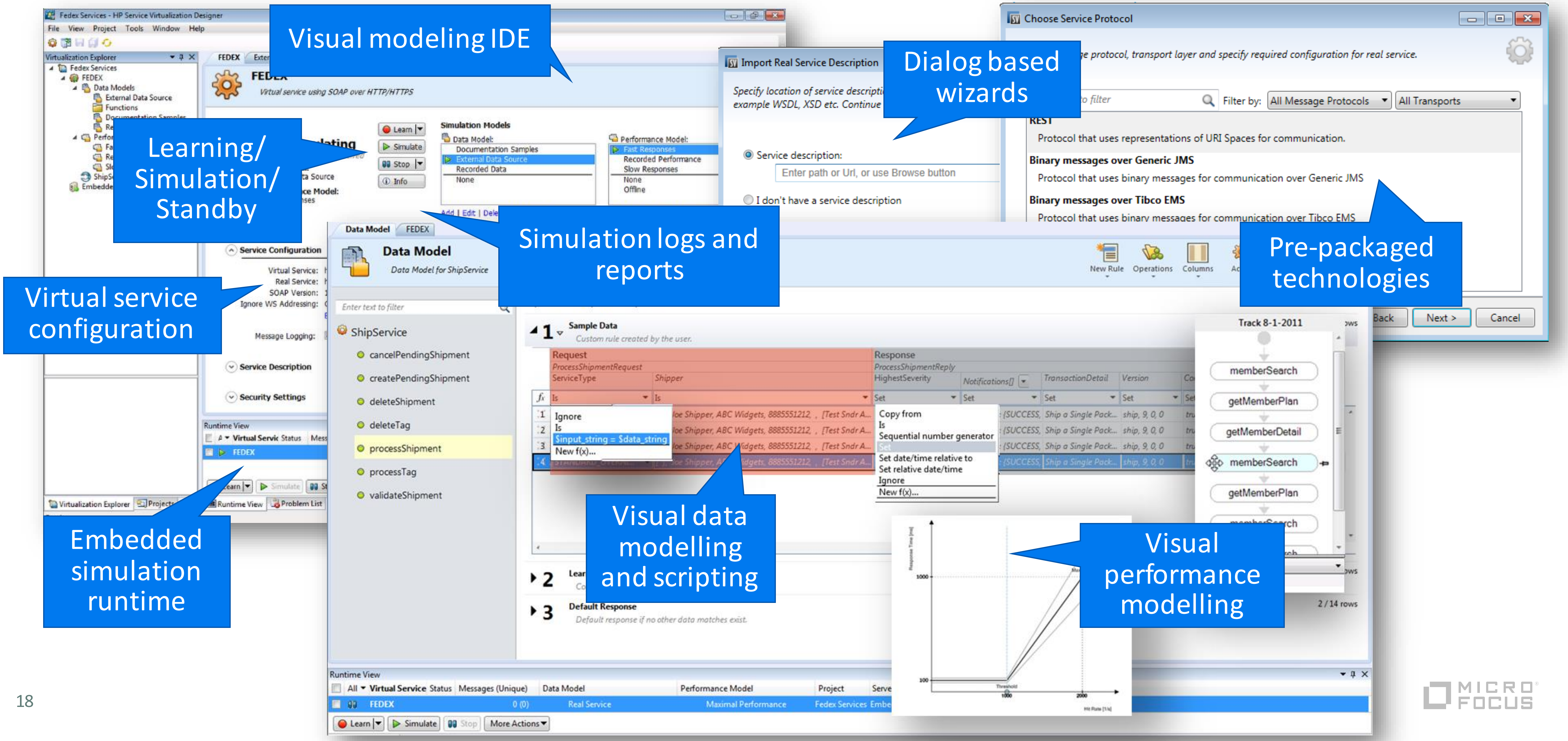


```
// (C) Copyright 2009-2019 Micro Focus or one of its affiliates.
import * as sv from "../../../../../../lib/sv-vsl.js"

export class HelloServiceModel extends sv.ServiceModel {

    constructor(service: sv.RestServiceInterface) {
        super(service);
        this.service = service;
    }

    @sv.scenario
    simpleSayHello() {
        this.service.GET( path: "/hello")
            .withRequest()                                             // empty request
            .withResponse({"greeting": "Hello, world!"}, sv.JSON)
                .withHeaders( headers: {"Content-Type": "application/json"})   // necessary response hea
                .withStatusCode( statusCode: 200);

        this.service.GET( path: "/hello")
            .withRequest()
            .withResponse({"greeting": "Hello again!"}, sv.JSON)
                .withHeaders( headers: {"Content-Type": "application/json"})
                .withStatusCode( statusCode: 200);
    }

}
```

> **“SV provides ease of use and enjoyable user experience and integrates with its own ALM testing tools like LoadRunner”**
>
> – Forrester Research, Service Virtualization Wave

- Simulate and invoke APIs and Virtual Services
- Introspect, Learn or Design from samples
- Support many protocols, message types and standards
- Define API, data and performance behavior profiles
- Stateless/Stateful/Asynchronous/Hybrid Simulation
- Publish/Subscribe and Test Driven simulation
- Learn and update dynamically as services change
- Visual simulation design, simulation language and scripting
- Data generation, pattern recognition, variables, auto-setup
- Desensitize data with Data Masking, drive from external data
- Reusable and shareable Virtual Service components
- Control and run from ADM test tools (UFT, LFT, Silk, LR, PC, MC)
- Simulation configuration and metrics included in Test Reports
- Build for automated, continuous testing and DevOps
- Run on Win, Linux and Docker
- Accessible and extensible with APIs and SV SDK

SV Designer - quick and easy virtualization



Visual modeling IDE

Dialog based wizards

Learning/ Simulation/ Standby

Simulation logs and reports

Pre-packaged technologies

Virtual service configuration

Visual data modelling and scripting

Visual performance modelling

Embedded simulation runtime

Fedex Services - HP Service Virtualization Designer

FEDEX

Virtual service using SOAP over HTTP/HTTPS

Import Real Service Description

Specify location of service description, for example WSDL, XSD etc. Continue

○ Service description:

Enter path or Url, or use Browse button

○ I don't have a service description

Choose Service Protocol

Choose protocol, transport layer and specify required configuration for real service.

Filter by: All Message Protocols | All Transports

REST

Protocol that uses representations of URI Spaces for communication.

Binary messages over Generic JMS

Protocol that uses binary messages for communication over Generic JMS

Binary messages over Tibco EMS

Protocol that uses binary messages for communication over Tibco EMS

Data Model

Data Model for ShipService

ShipService

- cancelPendingShipment
- createPendingShipment
- deleteShipment
- deleteTag
- processShipment
- processTag
- validateShipment

1 Sample Data

Custom rule created by the user.

2 Learned

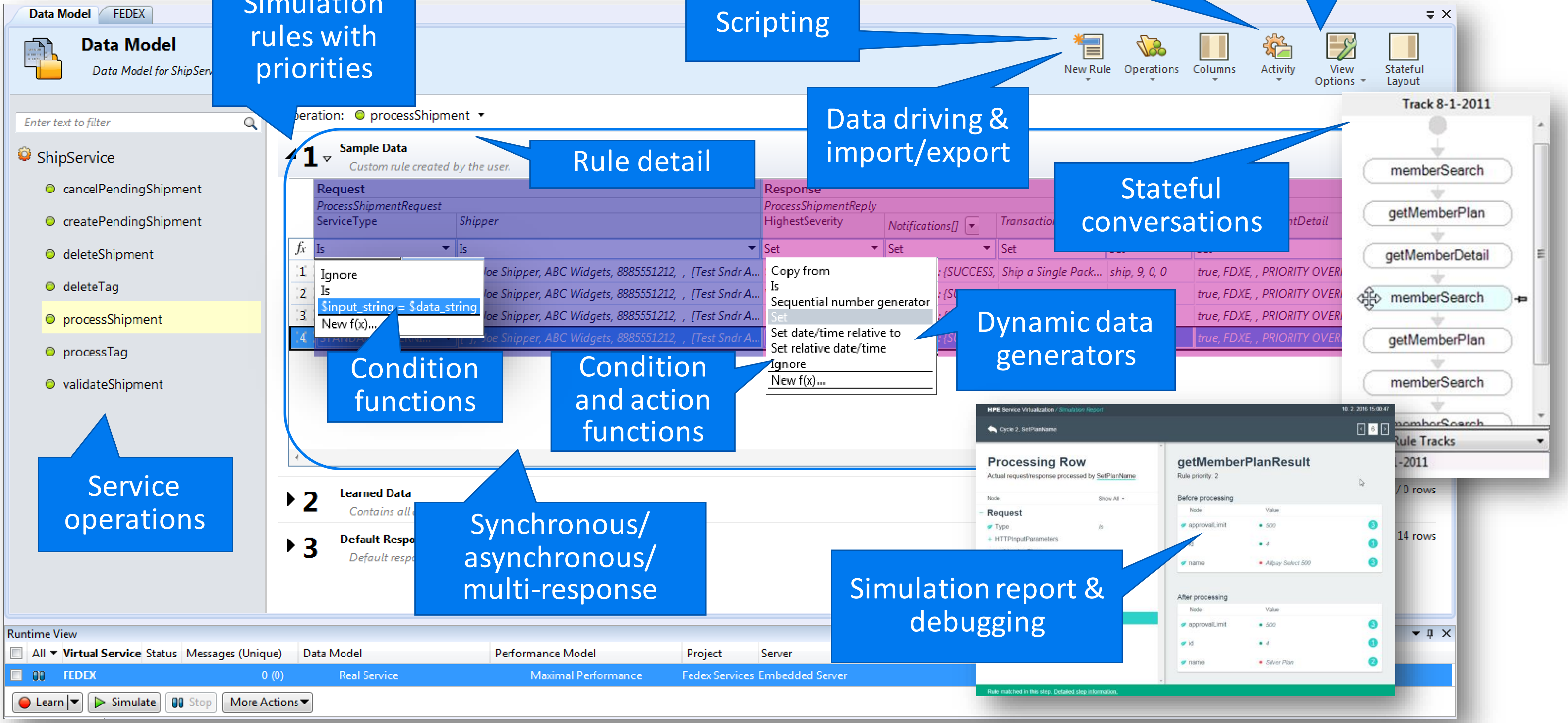
3 Default Response

Default response if no other data matches exist.

Runtime View

Virtual Service	Status	Messages (Unique)	Data Model	Performance Model	Project
FEDEX		0 (0)	Real Service	Maximal Performance	Fedex Services

Data Simulation and Functional Model



Simulation rules with priorities

Scripting

Hybrid simulation & external service communication

Filters & vertical data views

Rule detail

Data driving & import/export

Stateful conversations

Dynamic data generators

Condition functions

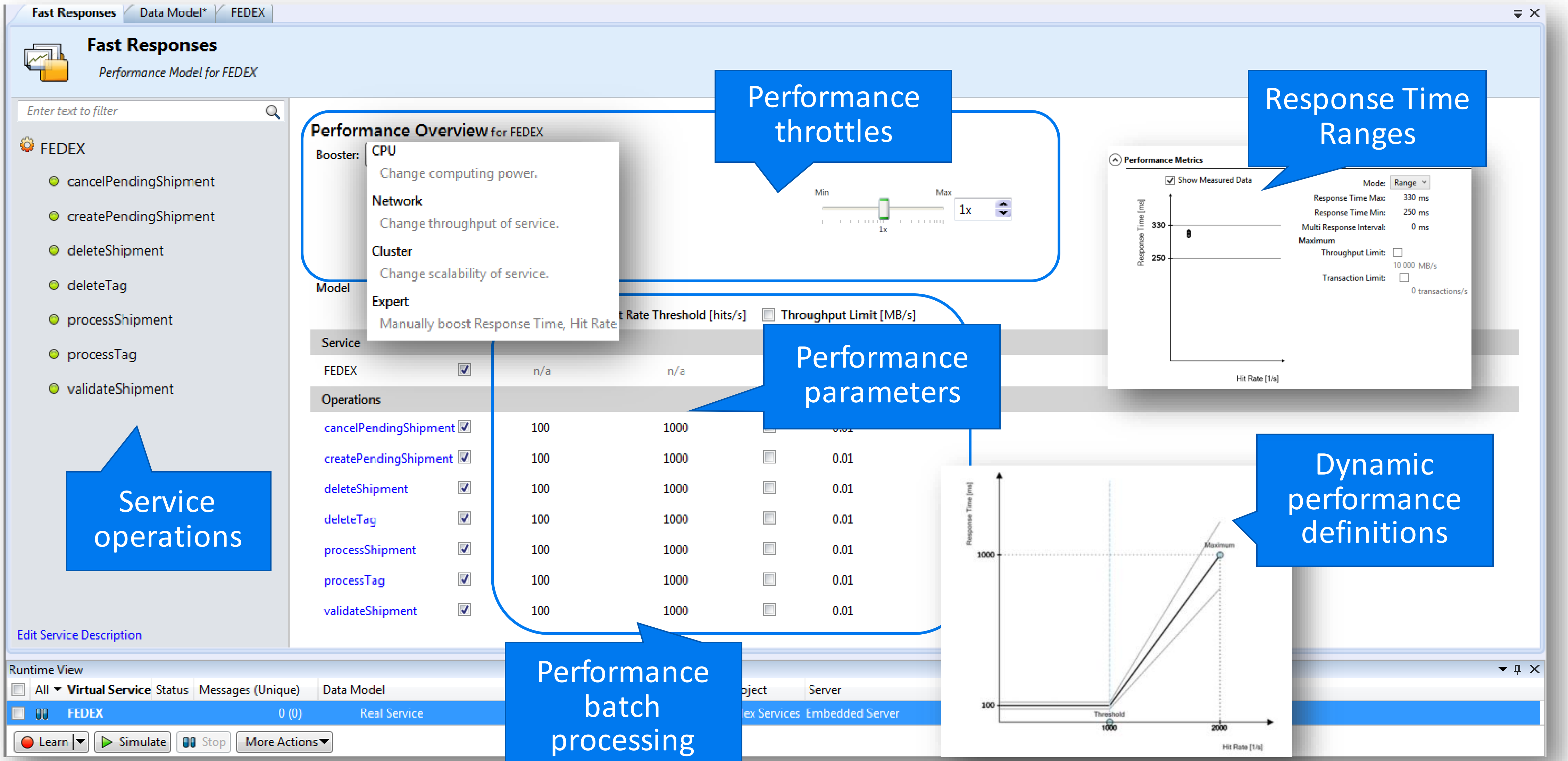
Condition and action functions

Service operations

Synchronous/ asynchronous/ multi-response

Simulation report & debugging

Performance Simulation and Model



Fast Responses

Performance Model for FEDEX

Performance Overview for FEDEX

Booster: CPU

- CPU: Change computing power.
- Network: Change throughput of service.
- Cluster: Change scalability of service.
- Expert: Manually boost Response Time, Hit Rate

Model

			t Rate Threshold [hits/s]	□ Throughput Limit [MB/s]	
Service					
FEDEX	☑	n/a	n/a		
Operations					
cancelPendingShipment	☑	100	1000	□	0.01
createPendingShipment	☑	100	1000	□	0.01
deleteShipment	☑	100	1000	□	0.01
deleteTag	☑	100	1000	□	0.01
processShipment	☑	100	1000	□	0.01
processTag	☑	100	1000	□	0.01
validateShipment	☑	100	1000	□	0.01

Performance Metrics

☑ Show Measured Data

Response Time [ms]: 330, 250; Hit Rate [1/s]

Mode: Range
Response Time Max: 330 ms
Response Time Min: 250 ms
Multi Response Interval: 0 ms
Maximum
Throughput Limit: □ 10 000 MB/s
Transaction Limit: □ 0 transactions/s

Runtime View

□ All	Virtual Service	Status	Messages (Unique)	Data Model	…ject	Server
□	FEDEX		0 (0)	Real Service	…ex Services	Embedded Server

Performance throttles

Response Time Ranges

Performance parameters

Service operations

Dynamic performance definitions

Performance batch processing

90+ options for SV covered technologies

TRANSPORT MESSAGE	HTTP(S) Gateway	HTTP(S) Proxy[1]	BLE GATT	NFC[2]	MQTT[2]	Java SDK	JMS	JDBC[1]	IBM WS MQ[1]	SAP RFC/XI/PI	TCP/IP	IMS Connect	CICS TS	Oracle AQ	Microsoft MQ	Web-Methods IS[1]	TIBCO AM/EMS[1]	JPOS	Flat Files/ FS/FTP/s
REST (XML, JSON, Bin)	✓	✓			✓											✓	✓		
XML*	✓	✓			✓		✓		✓	✓				✓	✓	✓	✓	✓	✓
Mobile (Andr., iOS)			✓	✓															
SOAP	✓	✓					✓		✓	✓				✓	✓	✓	✓		✓
Text/Binary	✓	✓					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
Java Objects						✓										✓			
AWS SQS[3]							✓												
SAP IDOC/APAB										✓						✓			✓
SWIFT MT/MX									✓							✓			
FIX									✓							✓			
ISO 8583[3]											✓					✓			
SQL								✓		✓				✓		✓	✓		
Cobol/Copybook	✓										✓	✓	✓			✓			
Fix Length									✓		✓					✓			
Text Delim., CSV																✓			✓

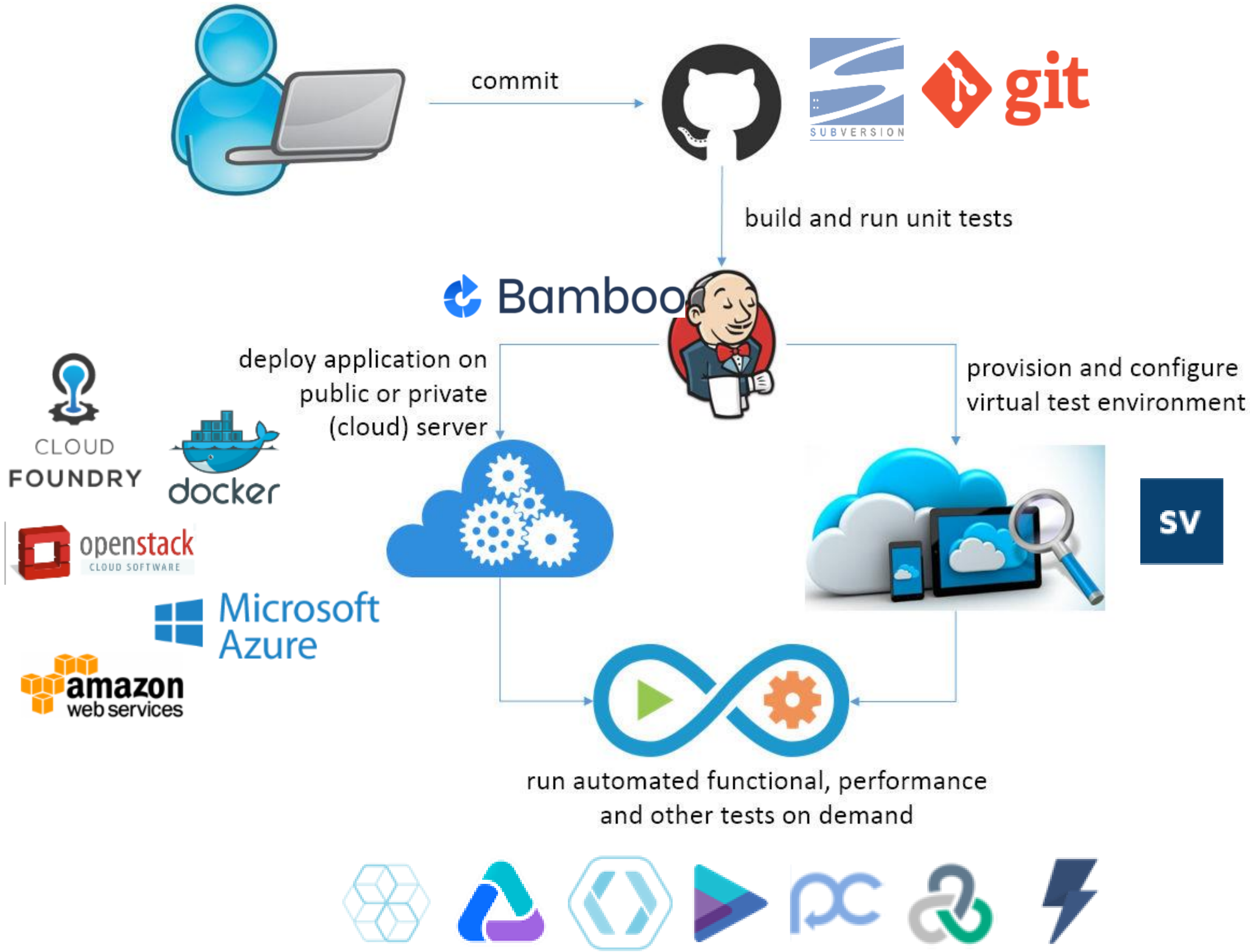
✓ Protocol supported

* All XML-based protocols supported

SV Protocol Extensibility SDK available

1 Non-intrusive
2 SV Lab only
3 Early Access

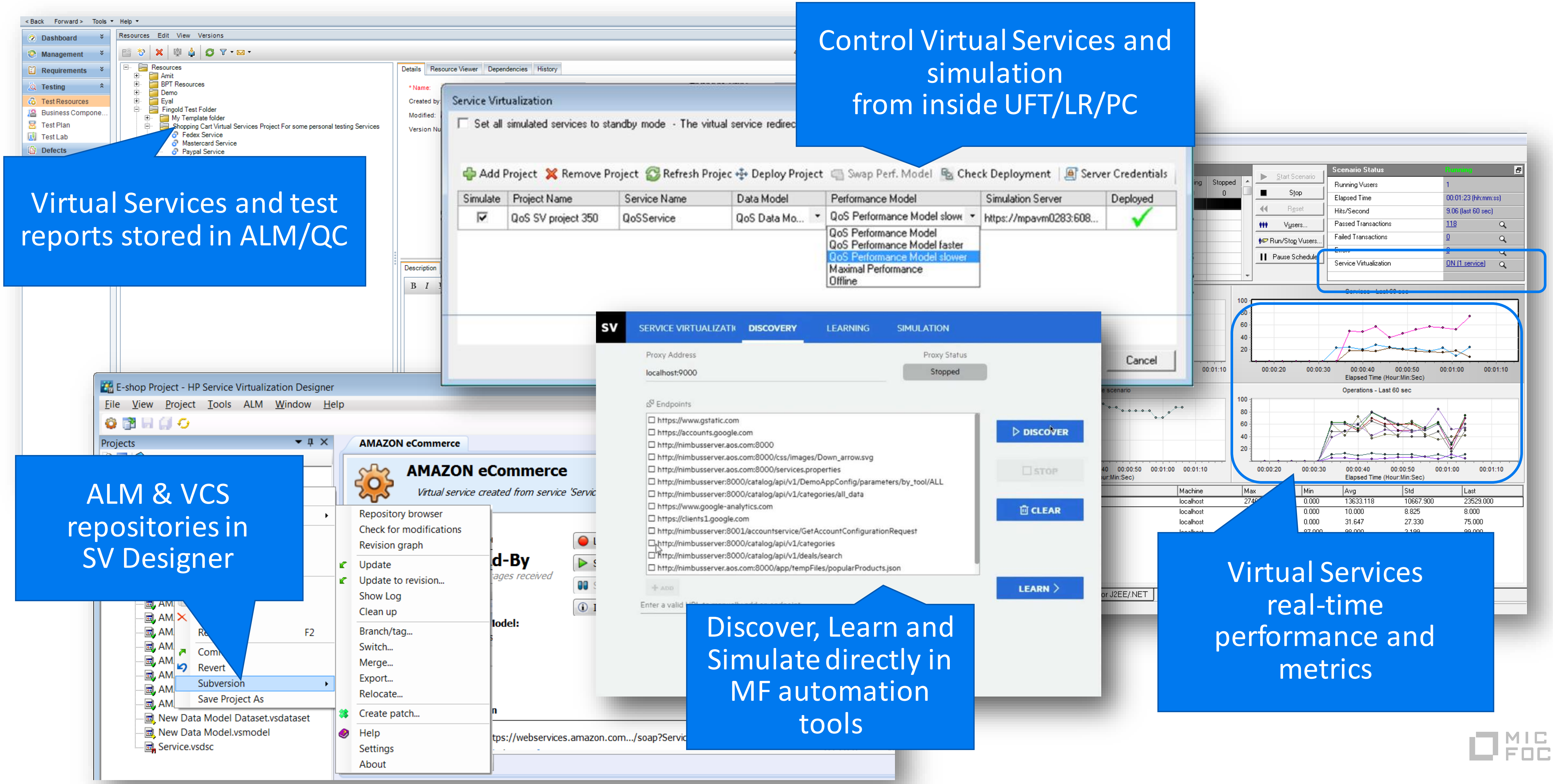
SV integration into continuous integration/delivery pipeline


commit
SUBVERSION
git
build and run unit tests
Bamboo
deploy application on public or private (cloud) server
provision and configure virtual test environment
CLOUD FOUNDRY
docker
openstack CLOUD SOFTWARE
Microsoft Azure
amazon web services
SV
run automated functional, performance and other tests on demand

- **Native Jenkins and Bamboo Plugins**
- **Dynamic Docker containers deployment**
- **Pre-integrated to ADM testing portfolio**
- **API and command-line tool for automation and integration with 3rd party tools**

- Application Automation Tools Jenkins Plugin
- App Delivery Management Bamboo CI Plugin

Pre-integrated with Micro Focus Test and ALM tools



Control Virtual Services and simulation from inside UFT/LR/PC

Virtual Services and test reports stored in ALM/QC

ALM & VCS repositories in SV Designer

Discover, Learn and Simulate directly in MF automation tools

Virtual Services real-time performance and metrics

Service Virtualization

☐ Set all simulated services to standby mode - The virtual service redirec

Simulate	Project Name	Service Name	Data Model	Performance Model	Simulation Server	Deployed
☑	QoS SV project 350	QoSService	QoS Data Mo...	QoS Performance Model slowe	https://mpavm0283:608...	✓

Proxy Address: localhost:9000

Proxy Status: Stopped

Endpoints

☐ https://www.gstatic.com
☐ https://accounts.google.com
☐ http://nimbusserver.aos.com:8000
☐ http://nimbusserver.aos.com:8000/css/images/Down_arrow.svg
☐ http://nimbusserver.aos.com:8000/services.properties
☐ http://nimbusserver:8000/catalog/api/v1/DemoAppConfig/parameters/by_tool/ALL
☐ http://nimbusserver:8000/catalog/api/v1/categories/all_data
☐ https://www.google-analytics.com
☐ https://clients1.google.com
☐ http://nimbusserver:8001/accountservice/GetAccountConfigurationRequest
☐ http://nimbusserver:8000/catalog/api/v1/categories
☐ http://nimbusserver:8000/catalog/api/v1/deals/search
☐ http://nimbusserver.aos.com:8000/app/tempFiles/popularProducts.json

Scenario Status: Running

Running Vusers	1
Elapsed Time	00:01:23 (hh:mm:ss)
Hits/Second	9.06 (last 60 sec)
Passed Transactions	118
Failed Transactions	0
Errors	0
Service Virtualization	ON (1 service)

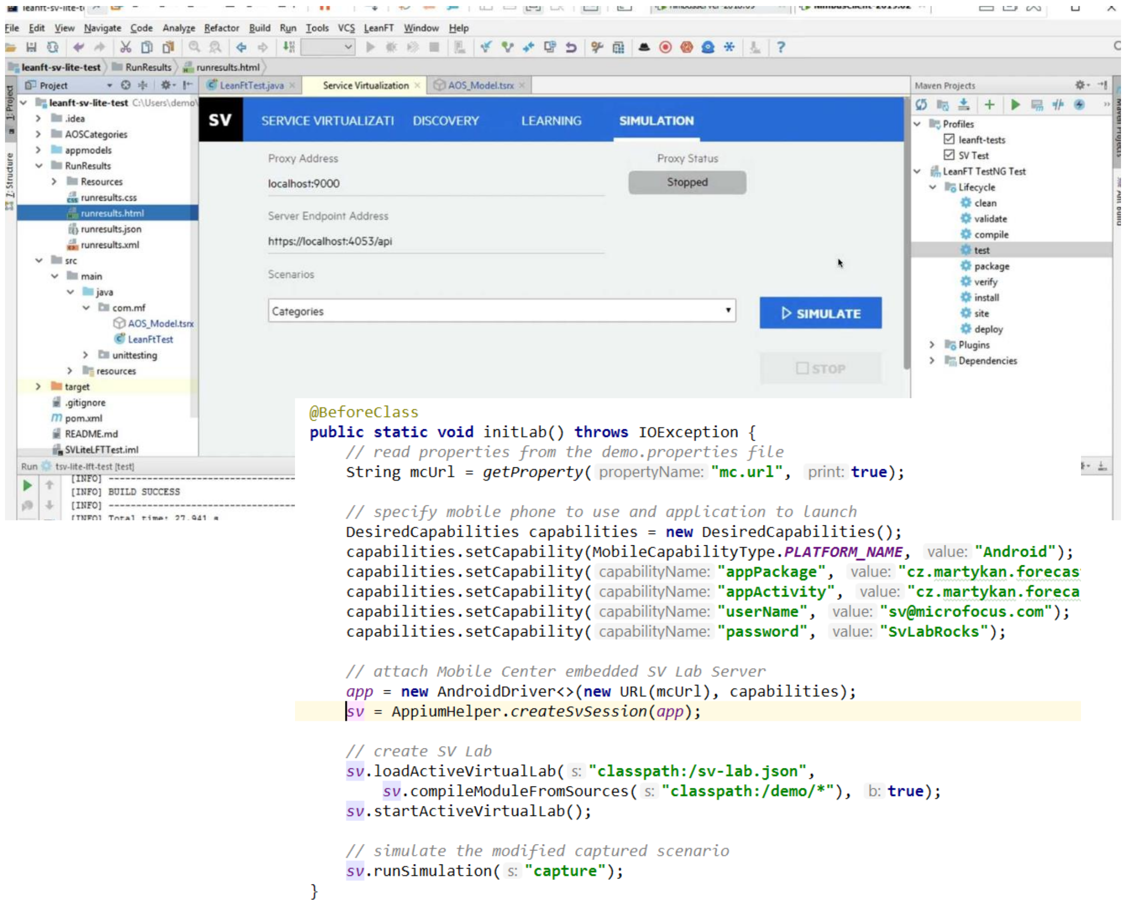
Machine	Max	Min	Avg	Std	Last
localhost	2749	0.000	13633.118	10667.900	23529.000
localhost		0.000	10.000	8.825	8.000
localhost		0.000	31.647	27.330	75.000

AMAZON eCommerce

Virtual service created from service 'Servic

Simulation inside of LeanFT and Silk4J

- SV Lab directly embedded and integrated into tools
- “Discover, Learn and Simulate” directly in the test automation tool without further SV installation
- Modify and store simulation assets within the test project
- Automate SV Lab provisioning and control scenarios from test script with SV Client library
- Direct access to simulation message log for debugging



Proxy Address: localhost:9000

Proxy Status: Stopped

Server Endpoint Address: https://localhost:4053/api

Scenarios: Categories

```
[INFO] ------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------
[INFO] Total time: 27.941 s
```

```
@BeforeClass
public static void initLab() throws IOException {
    // read properties from the demo.properties file
    String mcUrl = getProperty( propertyName: "mc.url",  print: true);

    // specify mobile phone to use and application to launch
    DesiredCapabilities capabilities = new DesiredCapabilities();
    capabilities.setCapability(MobileCapabilityType.PLATFORM_NAME,  value: "Android");
    capabilities.setCapability( capabilityName: "appPackage",  value: "cz.martykan.forecas
    capabilities.setCapability( capabilityName: "appActivity",  value: "cz.martykan.foreca
    capabilities.setCapability( capabilityName: "userName",  value: "sv@microfocus.com");
    capabilities.setCapability( capabilityName: "password",  value: "SvLabRocks");

    // attach Mobile Center embedded SV Lab Server
    app = new AndroidDriver<>(new URL(mcUrl), capabilities);
    sv = AppiumHelper.createSvSession(app);

    // create SV Lab
    sv.loadActiveVirtualLab( s: "classpath:/sv-lab.json",
        sv.compileModuleFromSources( s: "classpath:/demo/*"),  b: true);
    sv.startActiveVirtualLab();

    // simulate the modified captured scenario
    sv.runSimulation( s: "capture");
}
```

Early Access

Mobile Testing with Simulation

iOS

NFC

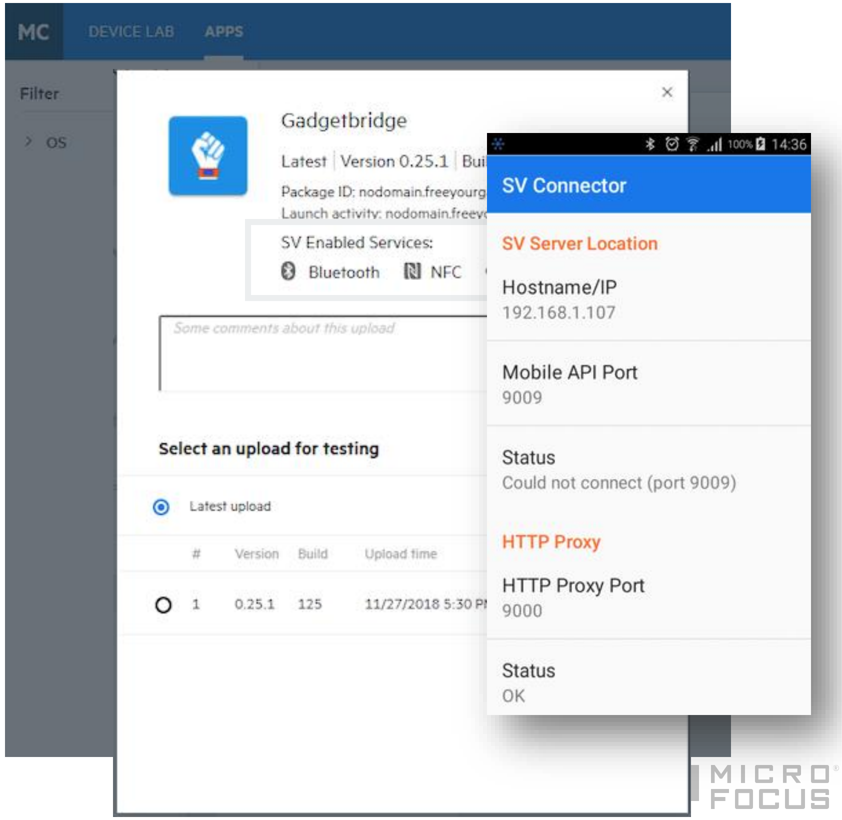
{ REST }

Bluetooth® SMART

Avoid physical manipulation with the mobile phone

Simulate different scenarios required by various tests

- SV lab deployed inside of MC ready to run simulations
- Ability to add instrumentation for SV for mobile apps during app upload
  - Android: HTTP/REST, BLE, NFC
  - iOS: HTTP/REST
- Automatic install of SV control on the mobile device (at the time of device plug in)
- Ability to turn SV on and off using SV control app on the mobile device



Gadgetbridge

Latest | Version 0.25.1 | Bui

Package ID: nodomain.freeyourg

Launch activity: nodomain.freey

SV Enabled Services:

Bluetooth NFC

Some comments about this upload

Select an upload for testing

◉ Latest upload

	#	Version	Build	Upload time
○	1	0.25.1	125	11/27/2018 5:30 PI

SV Connector

SV Server Location

Hostname/IP
192.168.1.107

Mobile API Port
9009

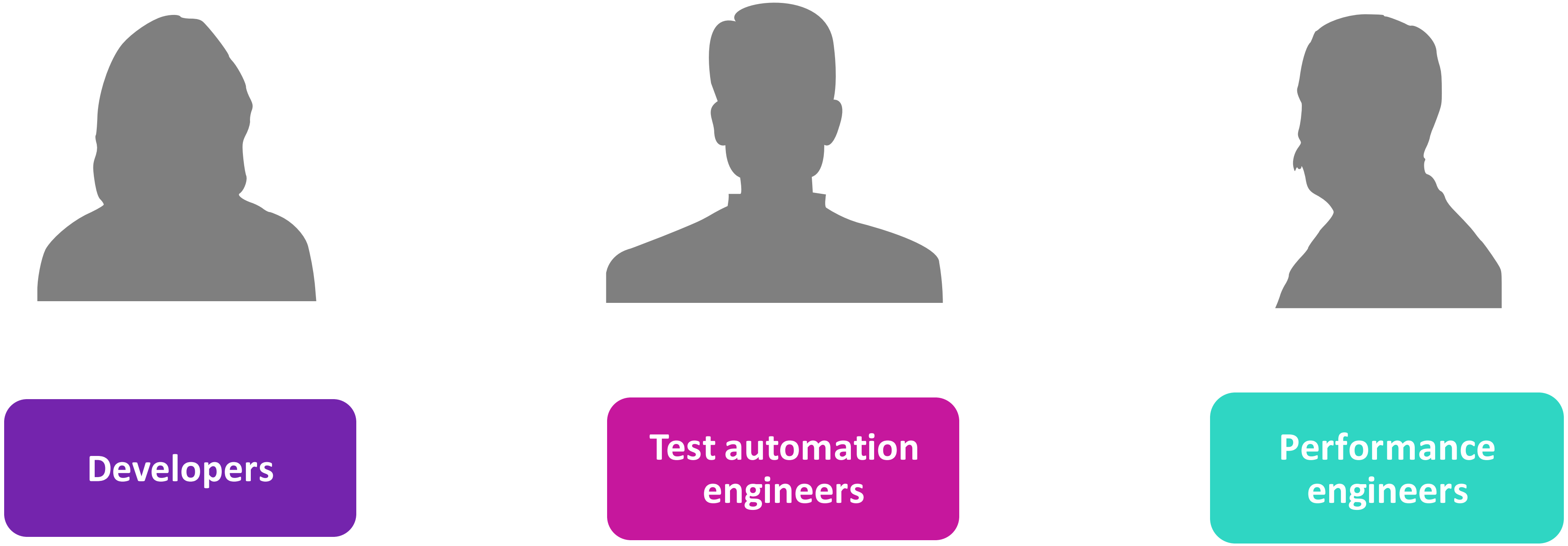
Status
Could not connect (port 9009)

HTTP Proxy

HTTP Proxy Port
9000

Status
OK

Service Virtualization for...



Developers

Test automation engineers

Performance engineers

Success Stories

Virgin Media ROI study

Proof of realized value (Sep 2016)

Prior situation/challenges

- **Non-availability of third party/legacy systems**
- **Delayed testing** Instability of downstream applications
- **Short testing windows** provided by other downstream systems
- Downstream applications availability for use **limited by the testing of production issues**
- **Integration testing challenges**

The journey

- Virtualized **70+ services** for functional and performance testing where did not have end to end test/performance environment for third party components
- Service Virtualization **enables testing to continue and more testing to be done**
- **Improved capabilities** to stress the applications to their limits and test resiliency

Quantified benefits (38 months)

- Cumulative benefits **£1,939,839**
- Cumulative investment **£196,667**
- **ROI 886%**
- Reduction in testing activity costs (£799k)
- Reduction in test environment build and related costs (£618k)
- Third party system cost savings (£523k)

Additional benefits not quantified

- **Faster time to bring products and services to market**
- **Reduction in defects** reaching production and associated **improvements in product quality**
- **Earlier detection of defects** through the ability to de-couple testing from application delivery schedules

http://files.asset.microfocus.com/3561/en/3561.pdf

Speeding up release cycles by eliminating testing delays for unlimited testing

Large internet service provider - IPTV, Voice over IP and mobile phone services

Challenge

- Deploy more than 200 apps per year. Major delays in testing and were never able to complete test on time. Test had a lot of down time.
- Test blocking issues were: Dependency on third party services, authorization issues, services not available during working hours, development was always behind.

Solution

- Virtualize 3rd party endpoints to validate coding changes
- Multiple development teams started using Service Virtualization to eliminate dependencies
- Modify performance factors of the virtualize service to ensure upstream calls
- Deploy virtual services as shared resource

Results

- Reduced test wait time 100%. On average saved 5-8 days/month.
- 95% of test cases are available through use of HP Service Virtualization.
- Increased test cycles and test coverage 45% - previously all test cases run in 112 hours comparing to just 54 hours with MF SV - 52% saving ($50K/month)
- ROI just in 3 months of SV use.

“We can start testing earlier without waiting the end to end systems to be ready. Now we can do efficient resource management, predict schedule and launch apps in a timely manner.”

Test and Release Manager,
TTNET

SAP Scalability and Performance Validation

Leading electronics retailer

Challenge

- Expensive stub solutions from development team which was expensive and slow
- Current stub solution had no integration to performance tools
- Need metrics on how the sub/virtual service was performing
- Staging data with their current process was some what time consuming

Solutions

- Service Virtualization SAP IDOC support expanded with Batch Performance Simulation
- Built in monitors on how virtual services are performing with out of the box integration to Performance Center
- Optimized SV throughput directly for SAP use cases and scenarios

Results

- MF SV processed ~2.5 times more IDOC messages per second than competition
- Service Virtualization selected as a tool for SAP functional and performance testing
- Used by local LOBs for functional integrations and central SAP Performance Test team
- Recently ran stress testing with 10s of millions of transactions per hour

> ***“We want to control our own destiny and not rely on stubs created by other departments so we can adapt and change faster.”***
>
> *SAP Performance Manager*

Proven ROI of using Service Virtualization

Across Application Projects, Operations and Customer Experience

25% ↑

Faster Time to Market

40% ↓

Fewer Production Defects

20% ↑

Improvement in Service Availability

75%

Reduced Wait Times

40%

Decreased Software Cycles

50%

Shorter Test and Defect Reproduction Times

2x

Increased Test Coverage

Source: Internal ROI Study 2017

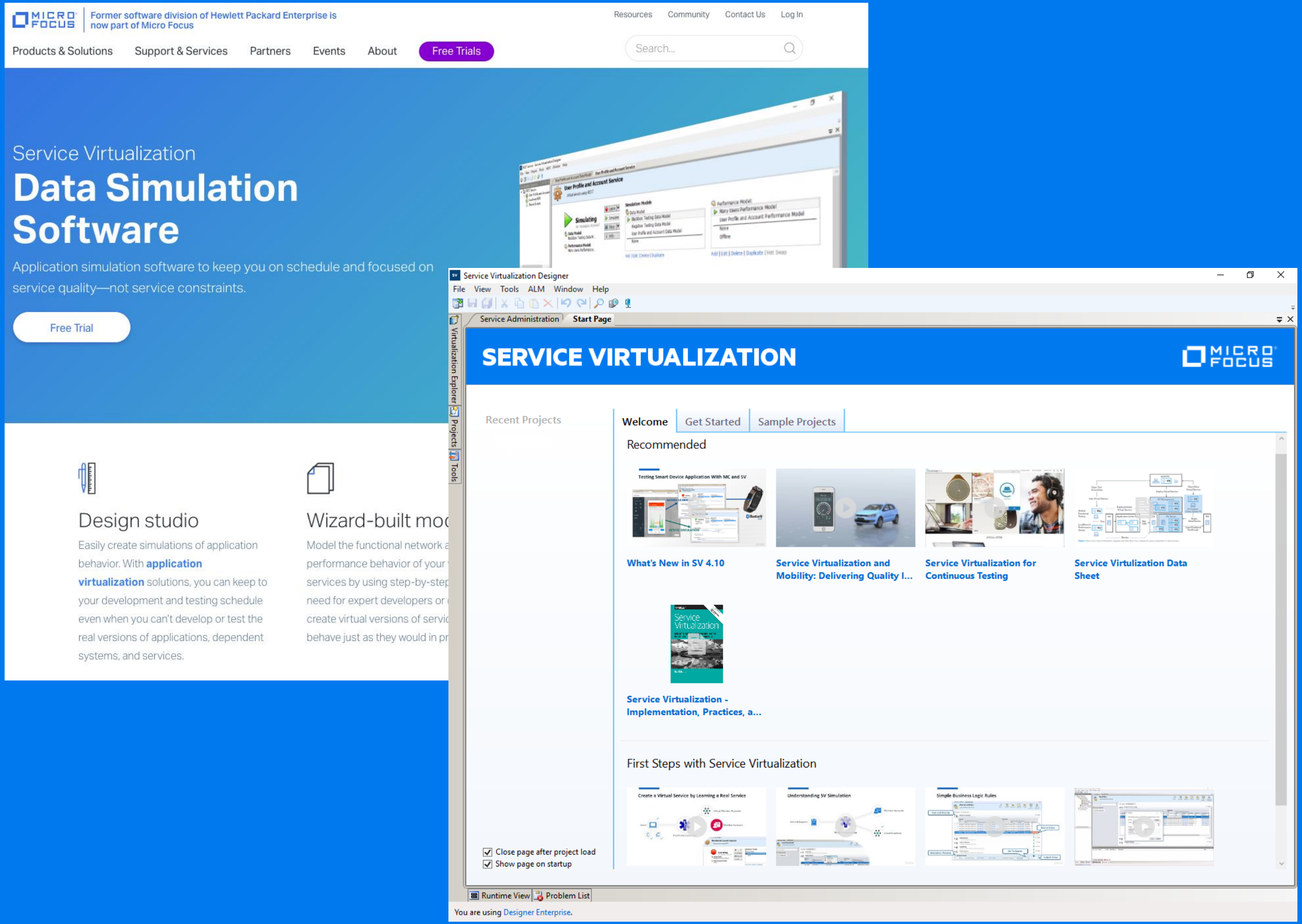
More public customer success stories?

http://www.microfocus.com/sv

SV Customer	Location
Virgin Media	http://files.asset.microfocus.com/3561/en/3561.pdf
SKY	https://www.microfocus.com/media/case-study/sky_cs.pdf
TTNET	http://www.briefingsdirectblog.com/2013/04/service-virtualization-brings-speed.html
EWTel	http://briefingsdirect.com/service-virtualization-solves-bottlenecks-amid-complex-billing-_process-for-german-telco
GameStop	http://files.asset.microfocus.com/4aa6-6125/en/4aa6-6125.pdf
Large EMEA Bank	https://www.microfocus.com/media/success-story/large_european_bank_ss.pdf
Patson USA	https://youtu.be/5T8obFkxrBg
Orasi	http://files.asset.microfocus.com/4aa6-6125/en/4aa6-6125.pdf

Get higher quality software to market in less time, for less cost with Service virtualization!

Faster turnaround


MICRO FOCUS
Former software division of Hewlett Packard Enterprise is now part of Micro Focus
Resources
Community
Contact Us
Log In
Products & Solutions
Support & Services
Partners
Events
About
Free Trials
Search...
Free Trial
Service Virtualization Designer
File
View
Tools
ALM
Window
Help
Service Administration
Start Page
Virtualization Explorer
Projects
Tools
MICRO FOCUS
Recent Projects
Welcome
Get Started
Sample Projects
Recommended
Testing Smart Device Application With MC and SV
What's New in SV 4.10
Service Virtualization and Mobility: Delivering Quality I...
Service Virtualization for Continuous Testing
Service Virtulization Data Sheet
Service Virtualization - Implementation, Practices, a...
First Steps with Service Virtualization
Create a Virtual Service by Learning a Real Service
Understanding SV Simulation
Simple Business Logic Rules
Runtime View
Problem List

Service Virtualization

Data Simulation Software

Application simulation software to keep you on schedule and focused on service quality—not service constraints.

Design studio

Easily create simulations of application behavior. With **application virtualization** solutions, you can keep to your development and testing schedule even when you can't develop or test the real versions of applications, dependent systems, and services.

Wizard-built moc

Model the functional network a performance behavior of your services by using step-by-step need for expert developers or create virtual versions of servic behave just as they would in pr

SERVICE VIRTUALIZATION

☑ Close page after project load
☑ Show page on startup

You are using Designer Enterprise.

www.microfocus.com/sv