
TOMORROW starts here.
cisco
Cisco live!

The Hitchhiker's Guide To onePK

BRKRST-2117

Richard Pruss

Principal Engineer

Cisco live!

"A new type of networking has come along that will change everything…"

Business Insider 4/11/12

“Don’t Panic!”

Douglas Adams

Agenda

- Why one Platform Kit?
- What's In the Kit?
- What Can You Do With onePK?
- Where does onePK fit in the SDN Universe?




Why One Platform Kit?

What is One Platform Kit?

OnePK is

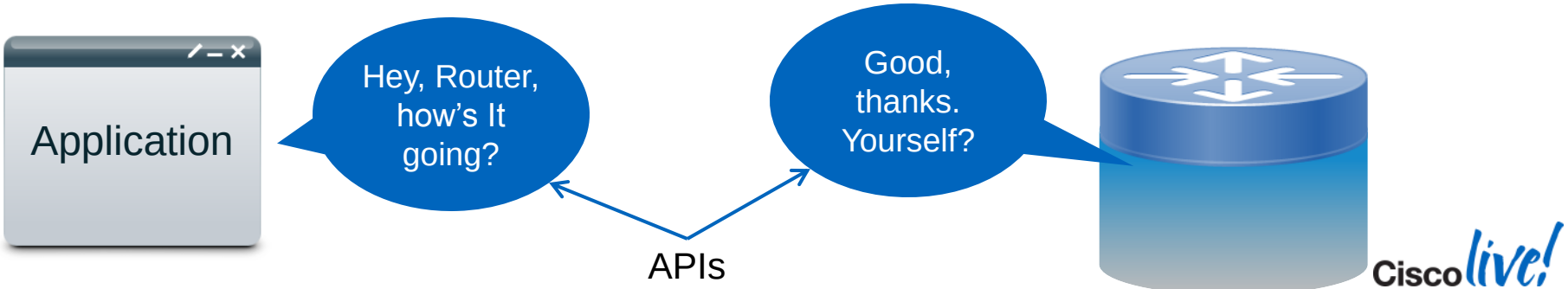
A software development kit (SDK) for Cisco Platforms

What Is An SDK?

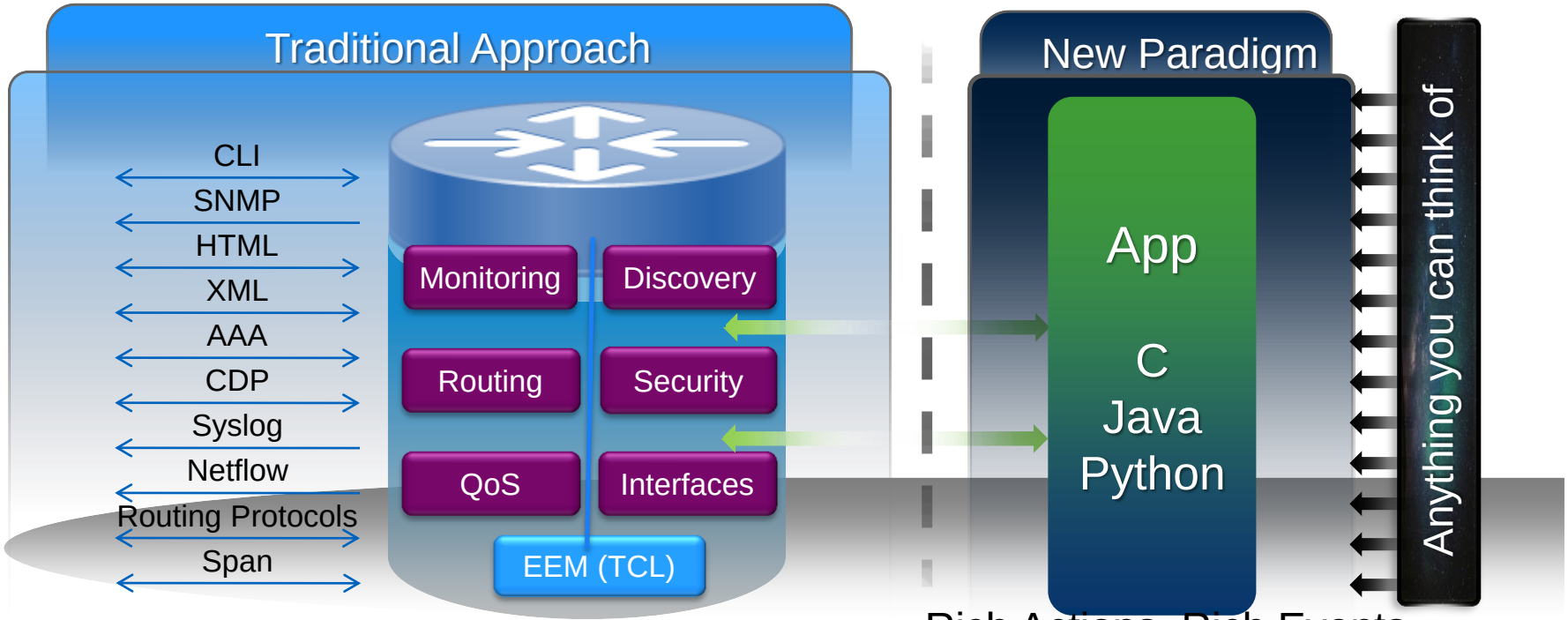
A Set of Application Programming Interfaces (API).

What Is an API?

Functions, variables, data structures that enable software components to communicate with each other.



Evolving How We Interact With Network Devices



Rich Actions, Rich Events,
Rich Environment

Introducing One Platform Kit - onePK

Flexible development environment to:

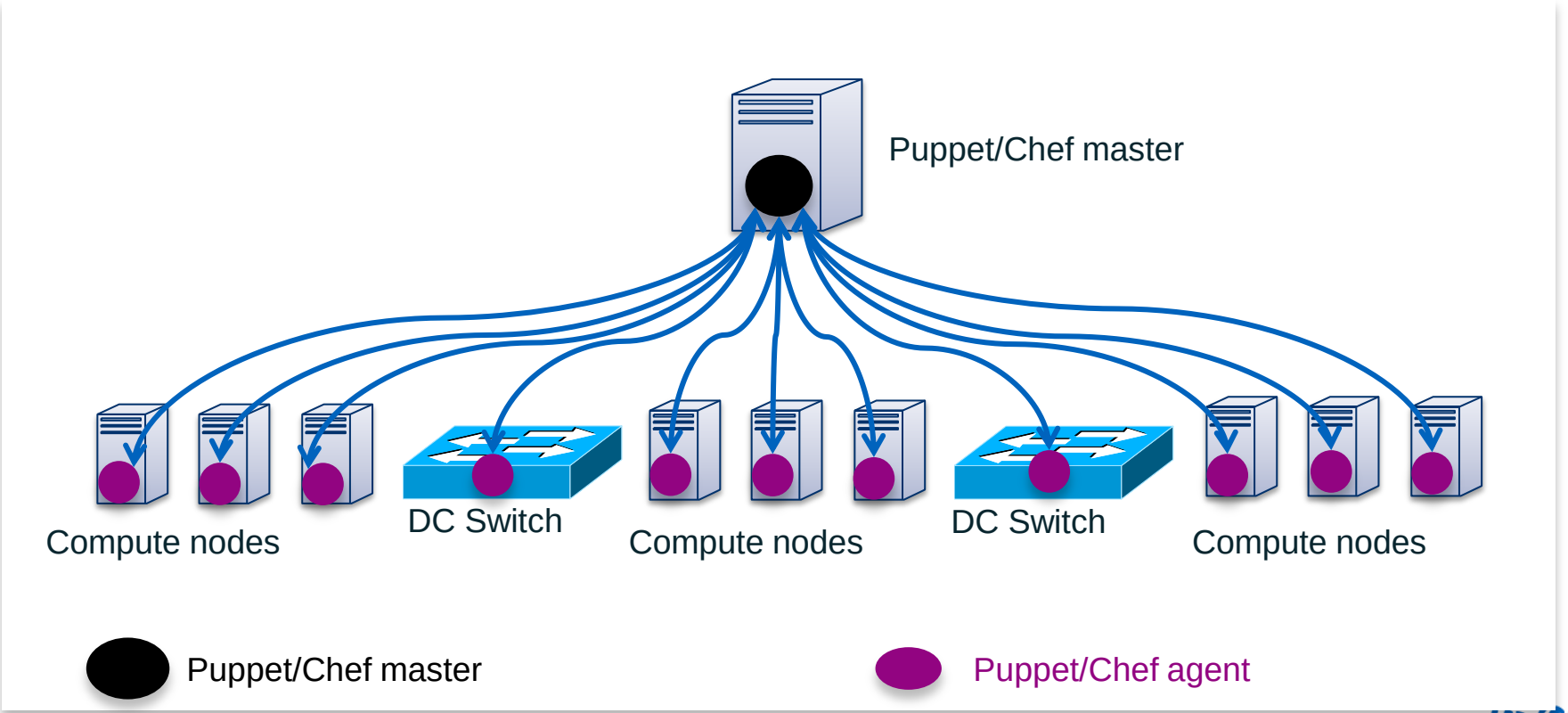
- Provision
- Monitor
- Maintain
- Troubleshoot
- Automate
- Innovate
- Extend

Simplify Operations

Enhance Agility

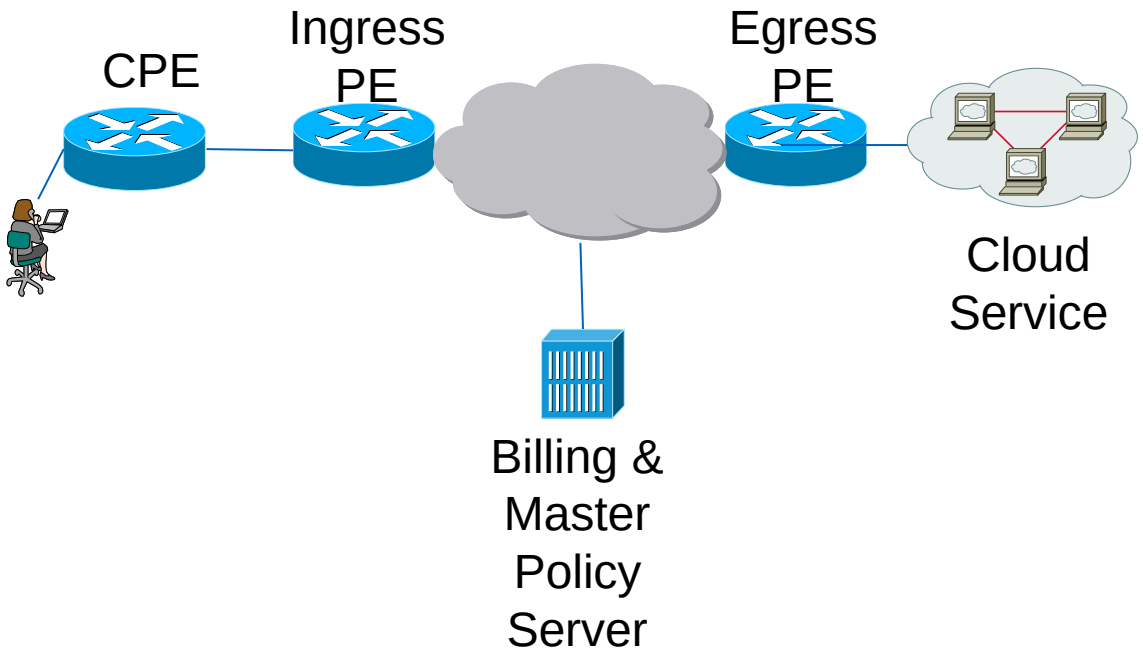
Revenue Up, OpEx Down

Simplify With The Tools You Know

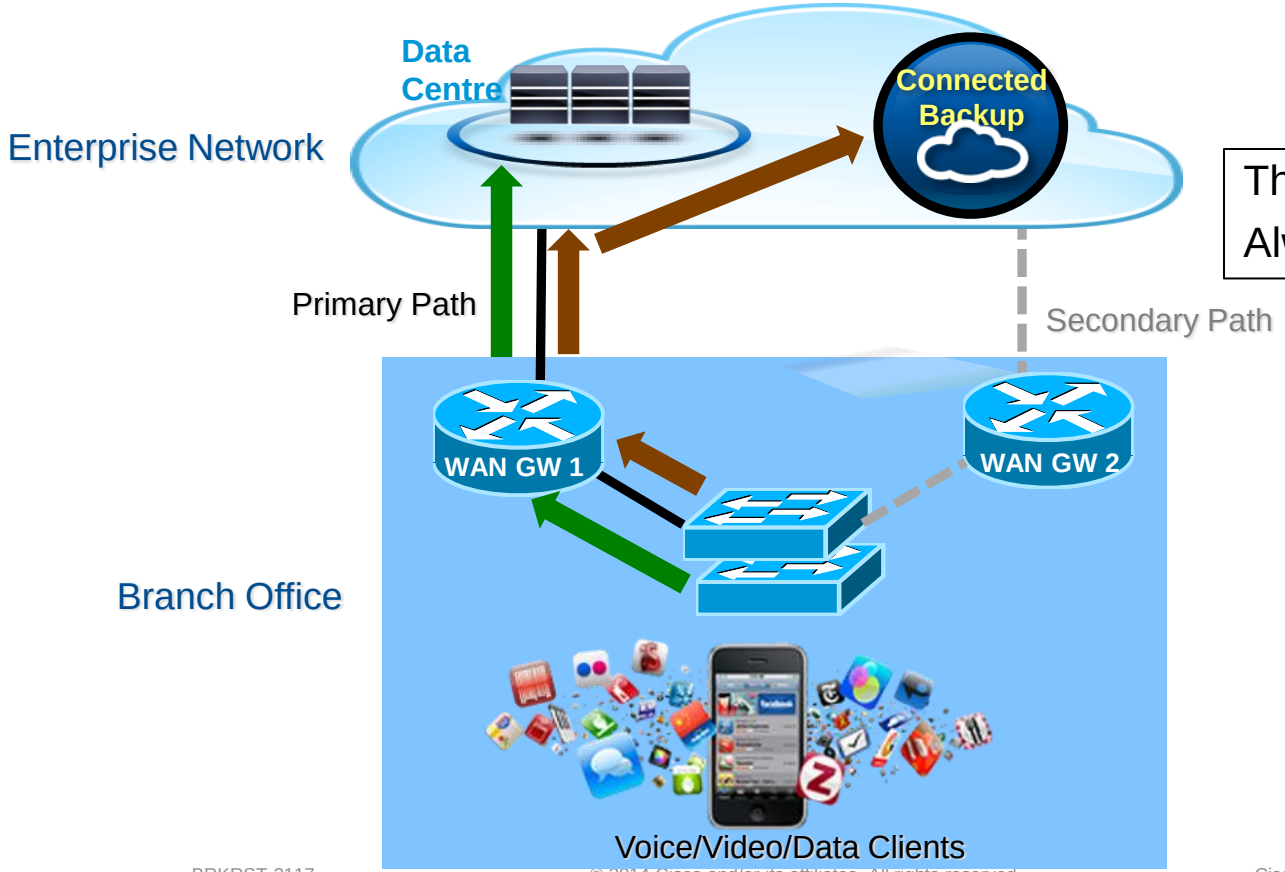


Revenue: Pay-as-You-Go QoS

1. Customer buys Pre-Pay QoS package for cloud service
2. First packet for new session appears on ingress PE and is relayed to Master server
3. Master server verifies pre-pay account and applies QoS
4. Ingress PE detects end of session and relays this to the Server
5. Server removes policy, bills customer for duration of session.

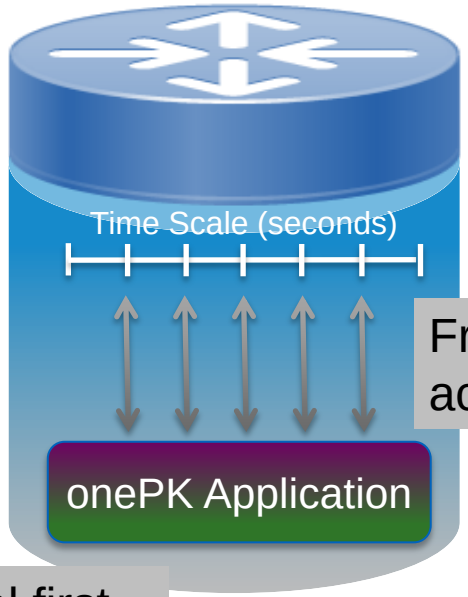


Reduce OpEx: Traffic Steering for Branch Offices



The High Availability Dilemma:
Always Paid For, Seldom Used

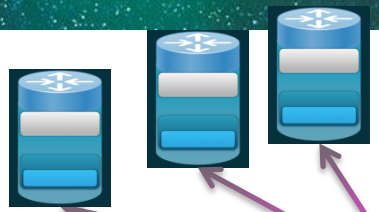
Agility: Network Be Nimble…



Local first-order analysis

Frequent local actions

Any communication protocol



Centralised coordination

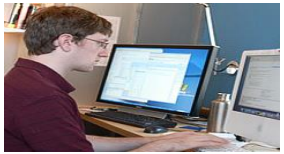
Centralised Management / Orchestration Application

Time Scale (minutes)

Consolidated central reporting.

Meta- and exception-analysis.

Do We Really Need Another Tool in the Tool Kit?



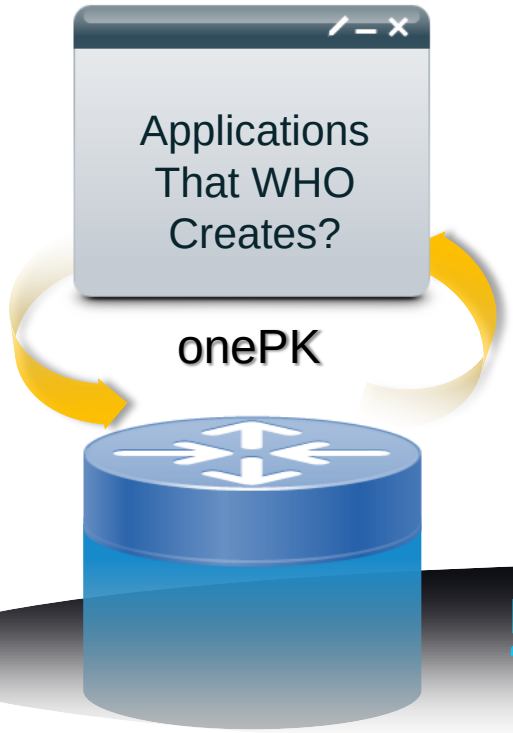
New Tools Enable New Users & New Use Cases

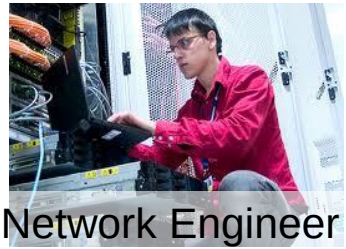
New Tools Enable Faster, Smarter Solutions

Seriously, Though, Why Programming Interfaces?

Simpler Interface	• Direct Socket Access • Explicit Error Checking
Application Friendly	• Speak Your Native Language
Integrated	• Many Data Sources • One Logic Stream

Who will be the Network Programmer?





Network, IOS Skills	■■■■■■■■■■
Scripting Skills	■■■■■□□□□□
Programming Skills	■■□□□□□□□□
Expertise	Network-centric use cases Scripts, PoCs, HA networks

Domain Expertise Will Remain Important
"Boundary" Skill Sets Will Grow

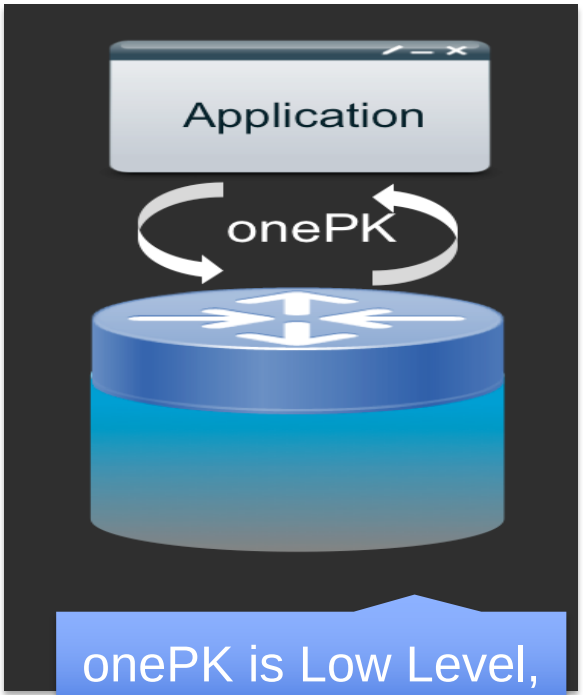
Top Three Reasons To Learn To Code (A Little)

It's Easier Than It Used to Be

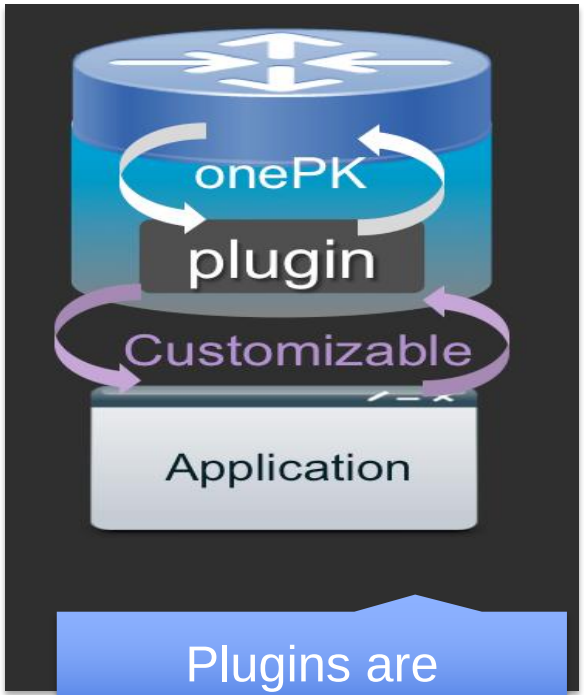
It's Useful

It's Fun

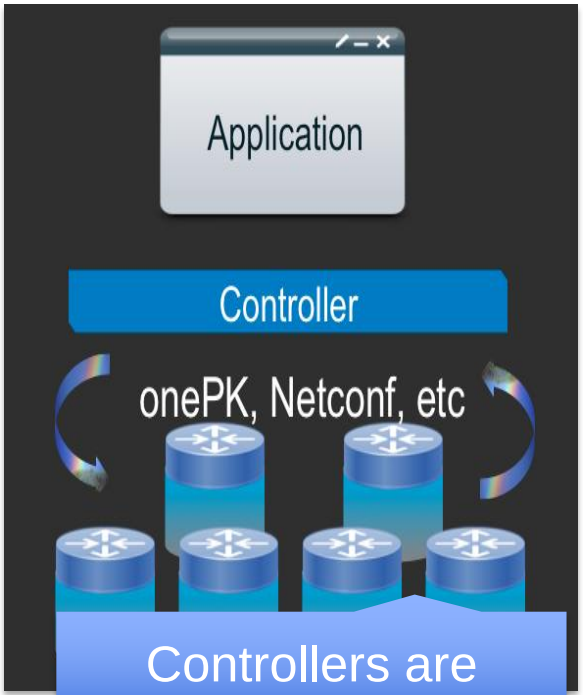
Top Three Reasons Not To Learn To Code (onePK)



onePK is Low Level, Device-Specific



Plugins are Adaptable

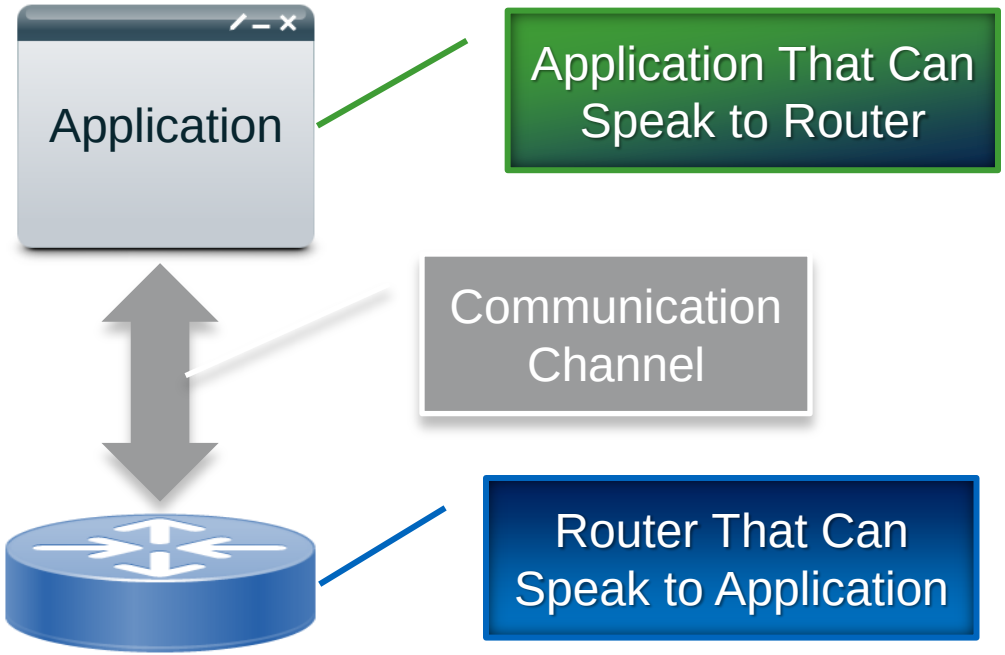


Controllers are Network Centric

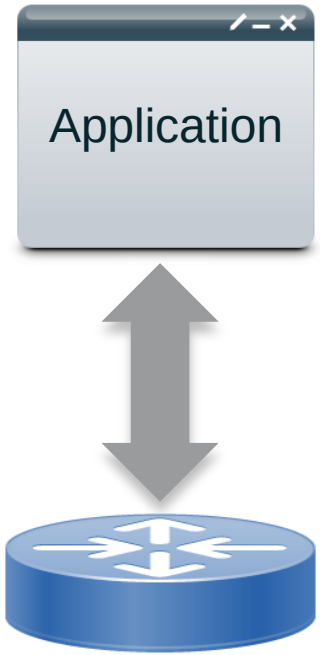



What's in the Kit?

onePK Architecture



onePK Architecture



C, JAVA, Python Program

onePK API Presentation

onePK API Infrastructure

IOS / XE (Catalyst, ISR, ASR1K)

NXOS (Nexus Platforms)

IOS XR (ASR 9K, CRS)

What Does the Network Abstraction Layer "Look Like"?

```
fred#dir
Directory of flash0:/
    1  -rw-    98356780   Nov 5 2013 23:23:00 +00:00  c2951-universalk9-
                                                      mz.SSA.154-1.8.T


fred(config)#username app1 password pass1
fred(config)#onep
fred(config-onep)#transport type tls disable-remotecert-validation
```

How to See if the API Infrastructure is Enabled

fred#**show onep status**

Status: enabled

Version: 1.1.0

Transport: tls; Status: running; Port: 15002

Transport: tipc; Status: disabled

Session Max Limit: 10

CPU Falling Rising Threshold: 0%

…

How to See if an Application is Connected

fred#**show onep statistics session all**

Session ID: 5671

Application Name: RIBTutorial

API In: 3 API Out: 1

Bytes In: 518 Bytes Out: 1961

Memory Allocated: 91256 bytes Memory Freed: 85080 Memory Held: 31312

CPU utilisation for five seconds: 0.0 % one minute: 0.13% five minutes: 0.3 %

fred#

You Have the Power To Stop Applications

```
fred#show onep session all
ID       Username State          ReconnectTimer ConnectTime              ApplicationName
305      user1    Connected      0              Sun Nov 24 00:28:40.948  RIBTutorial
fred#onep stop session ?
  RIBTutorial-305  Application name: RIBTutorial, Session ID: 305
  all              All sessions
fred#onep stop session all
fred#show onep session all
fred#
```

Keeping An Eye On Applications

fred#**show onep history session all**

Active Session: History Entry Count: [8]

[11/24/13 00:33:42.519 1] **createNamedAcl**_IDL(597,1,0,1)

[11/24/13 00:33:42.543 2]**addNamedL3Ace**_IDL('onep-acl-597-4',0,0,1,(10,0,1,'0.0.0.0',0,'0.0.0.0',0,6,0x0,0,0,0x0,0,0,34,27,2,0x0,16,0,597,597,0))

[11/24/13 00:33:42.651 3]**NetworkElement_getInterfaceList**IDL(597,-1,1,0,0)

[11/24/13 00:33:43.161 8]**applyNamedAclToInterface**_IDL('onep-acl-597-4',1,2,2,1)

fred#**show ip access-list dynamic**

Extended IP access list onep-acl-597-4

    10 deny tcp any any match-all -ack -fin -psh +syn log

fred#

What Does the C Presentation Layer "Look Like"?

C Program

onePK API Presentation

```
cisco@onepk:~$ ls /opt/cisco/onep/c64/sdk-c64-1.1.0.52/c
bin  doc  include  lib  README  sample-apps  tutorials  tutorials-html
cisco@onepk:~$ ls /opt/cisco/onep/c64/sdk-c64-1.1.0.52/c/lib/*.so
lib/libonep64_datapath.so   lib/libonep_core.so
…
cisco@onepk:~$ ls /opt/cisco/onep/c64/sdk-c64-1.1.0.52/c/include
onep_aaa.h                          onep_location_change_filter.h
onep_acl.h                          onep_location_constants.h
…
```

A C Application Uses the Presentation Layer

C Program

onePK API Presentation

```
cisco@onepk:$ grep onep_acl ACLTutorial.c
#include "onep_acl.h"
    onep_acl_t*              aclA;
    rc = onep_acl_create_l3_acl(AF_INET, ne, &aclA);
    rc = onep_acl_create_l3_ace(1, FALSE, &ace1);
    rc = onep_acl_set_l3_ace_protocol(ace1, ONEP_PROTOCOL_TCP);
    rc = onep_acl_set_l3_ace_src_prefix(ace1, NULL, 24);
…
```

header file

Compile, Link and Run

```
cisco@onepk:$ make
Building file: ACLTutorial.c
Invoking: GCC C Compiler
cc -I/opt/cisco/onep/c64/sdk-c64-1.1.0.52/c/include -Wall -c -fPIE -O1 -m64 -MMD –MP –MF"obj/ACLTutorial.d" -MT"obj/ACLTutorial.d" -o"obj/ACLTutorial.o” "ACLTutorial.c"
Finished building: ACLTutorial.c
Building target: bin/ACLTutorial
Invoking: GCC C Linker
cc -o bin/ACLTutorial obj/ACLTutorial.o -m64 –pie -L/opt/cisco/onep/c64/sdk-c64-1.1.0.52/c/lib -lonep64_core -lrt
Finished building target: bin/ACLTutorial
cisco@onepk:$ bin/ACLTutorial –a 10.10.10.130
Connected to network element: 10.10.10.130
…
```

Using the Java Presentation Layer

Java Program

onePK API Presentation

```
cisco@onepk:~$ ls /opt/cisco/onep/java/sdk-java-1.1.0.52/java
doc  lib  README.maven_jars  sample-apps  tutorials  tutorials-html
cisco@onepk:~$ ls /opt/cisco/onep/java/sdk-java-1.1.0.52/java/lib/
libonep-core-rel-1.1.0.52.jar              /java/lib/libonep-core-rel.jar

cisco@onepk:~$ mvn install:install-file -Dfile=libonep-core-rel.jar -
DgroupId=com.cisco.onep -DartifactId=libonep-core-rel -Dpackaging=jar
```

"The best drink in existence is the Pan Galactic Gargle Blaster. The effect of a Pan Galactic Gargle Blaster is like having your brains smashed out by a slice of lemon wrapped round a large gold brick.

Douglas Adams, Hitchhiker's Guide to the Galaxy

Demo: The Python Presentation Layer

“*Harmless*! Is that all it's got to say? *Harmless*! One word! …I hope you managed to rectify that a bit.”

"Oh yes, well I managed to transmit a new entry off to the editor. He had to trim it a bit, but it's still an improvement."

"And what does it say now?" asked Arthur.

"*Mostly harmless*," admitted Ford with a slightly embarrassed cough.

onePK Functions are Grouped in Service Sets

Base Service Set	Description
Data Path	Provides packet delivery service to application: Copy, Punt, Inject
Policy	Provides filtering (ACL), classification (Class-maps, Policy-maps), actions (Marking, Policing, Queuing, Copy, Punt) and applying policies to interfaces on network elements
Routing	Read RIB routes, add/remove routes, receive RIB notifications
Element	Get element properties, CPU/memory statistics, network interfaces, element and interface events
Discovery	L2 topology and local service discovery
Utility	Syslog events notification, Path tracing capabilities (ingress/egress and interface stats, next-hop info, etc.)
Developer	Debug capability, CLI extension which allows application to extend/integrate application's CLIs with network element

Where do onePK Applications Run?

Choose the Hosting Model that Suits Your Platform and Your Application

On An External Server

- Plentiful memory/compute
- Higher latency and delay
- Supported on by all platforms

"End-Node"

On A Hardware Blade

- Dedicated memory/compute
- Low latency and delay
- Requires modular hardware blade

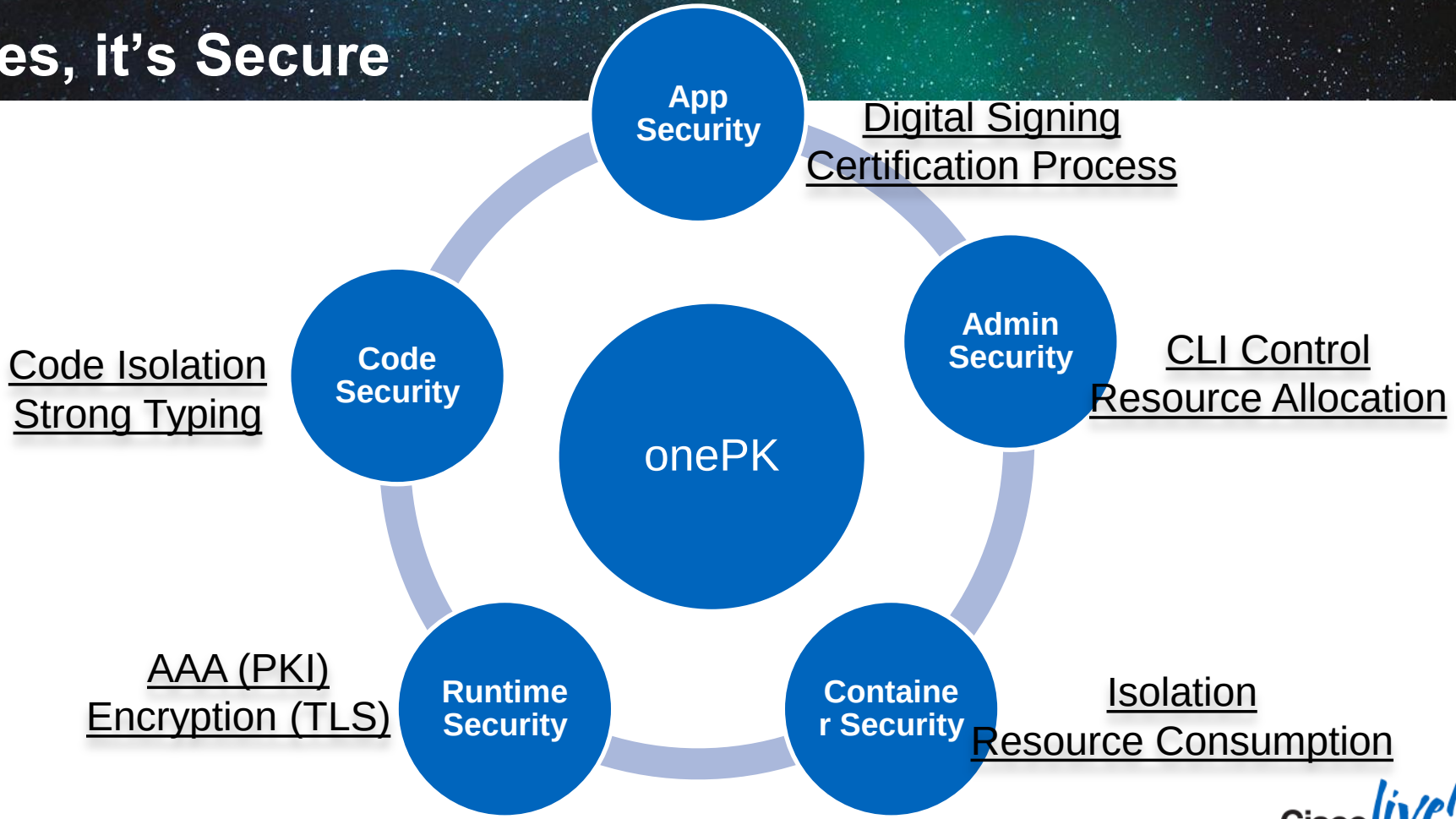
"Blade"

On the Router

- Shared memory/compute
- Very low latency and delay
- Available on select platforms

"Process"

Yes, it's Secure

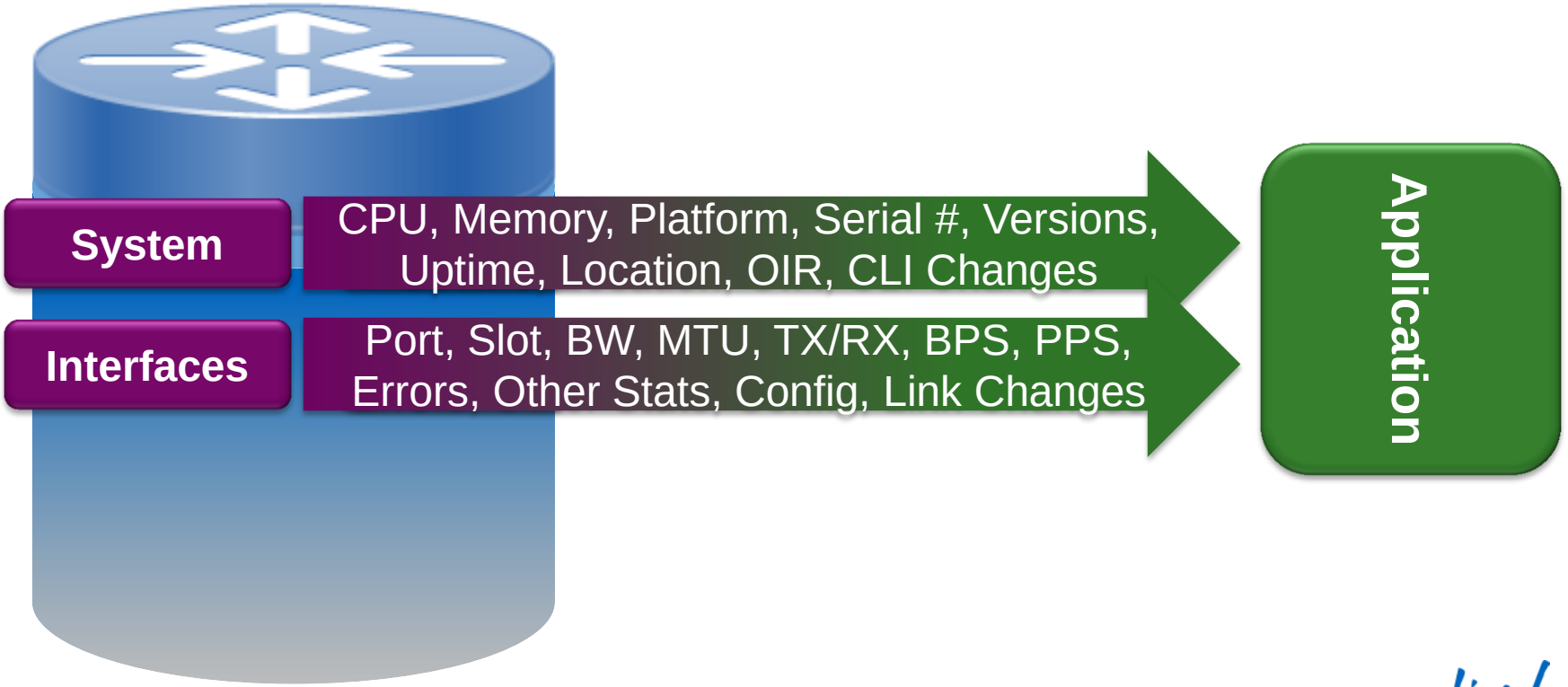





What can you Build?

What Could You Do If You Could…

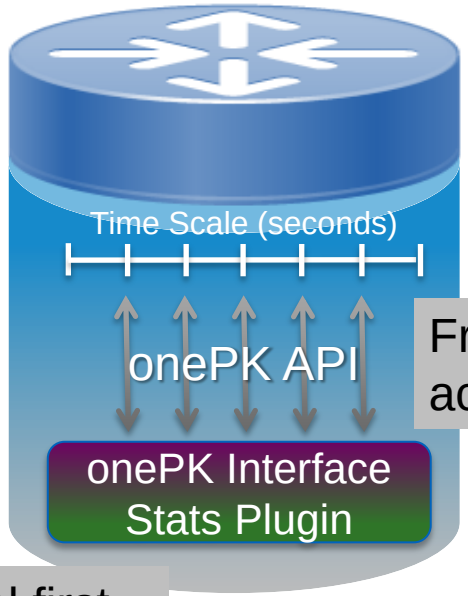
Get System and Interface Info?



Demo: “Hello Router” and other events

“One of the things Ford Prefect had always found hardest to understand about humans was their habit of continually stating and repeating the very, very obvious.”
— Douglas Adams, The Hitchhiker's Guide to the Galaxy

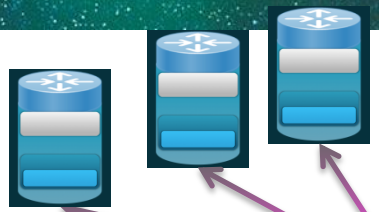
Interface Monitoring “Plugin”



Local first-order analysis

Frequent local actions

Any communication protocol (xmpp, OF, CIM, REST, etc)



Centralised coordination

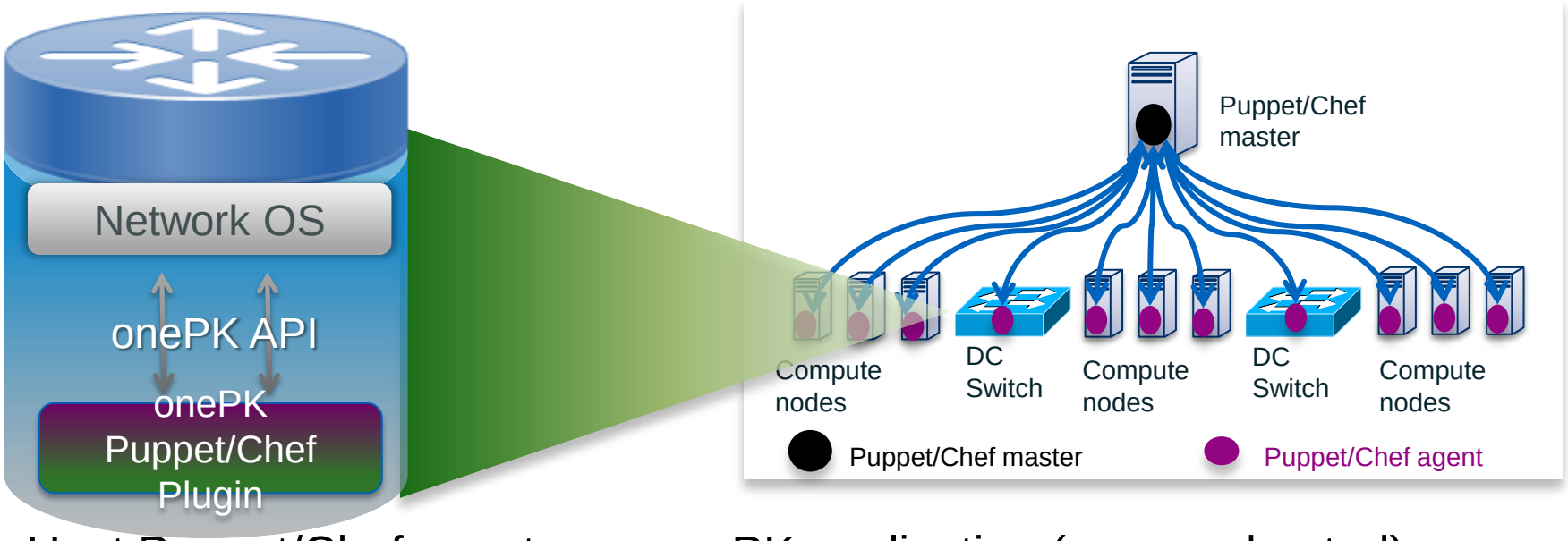
Centralised Management / Orchestration Application

Time Scale (minutes)

Consolidated central reporting.

Meta- and exception-analysis.

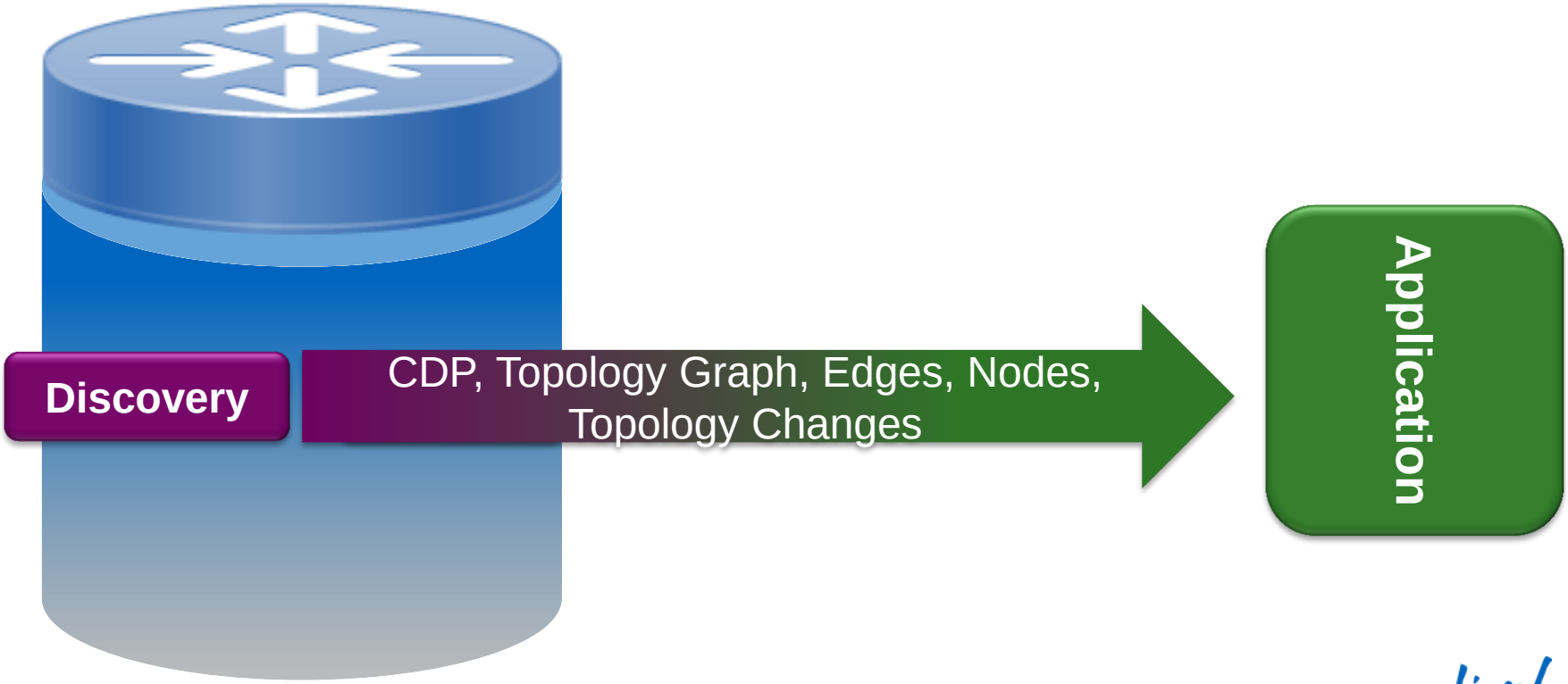
And Speaking of Plugins…



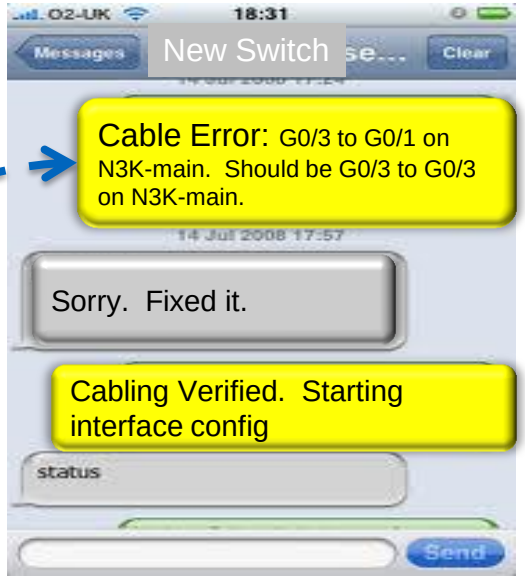
- Host Puppet/Chef agent as a onePK application (process hosted)
- Use onePK configuration API to implement configuration change

What Could You Do If…

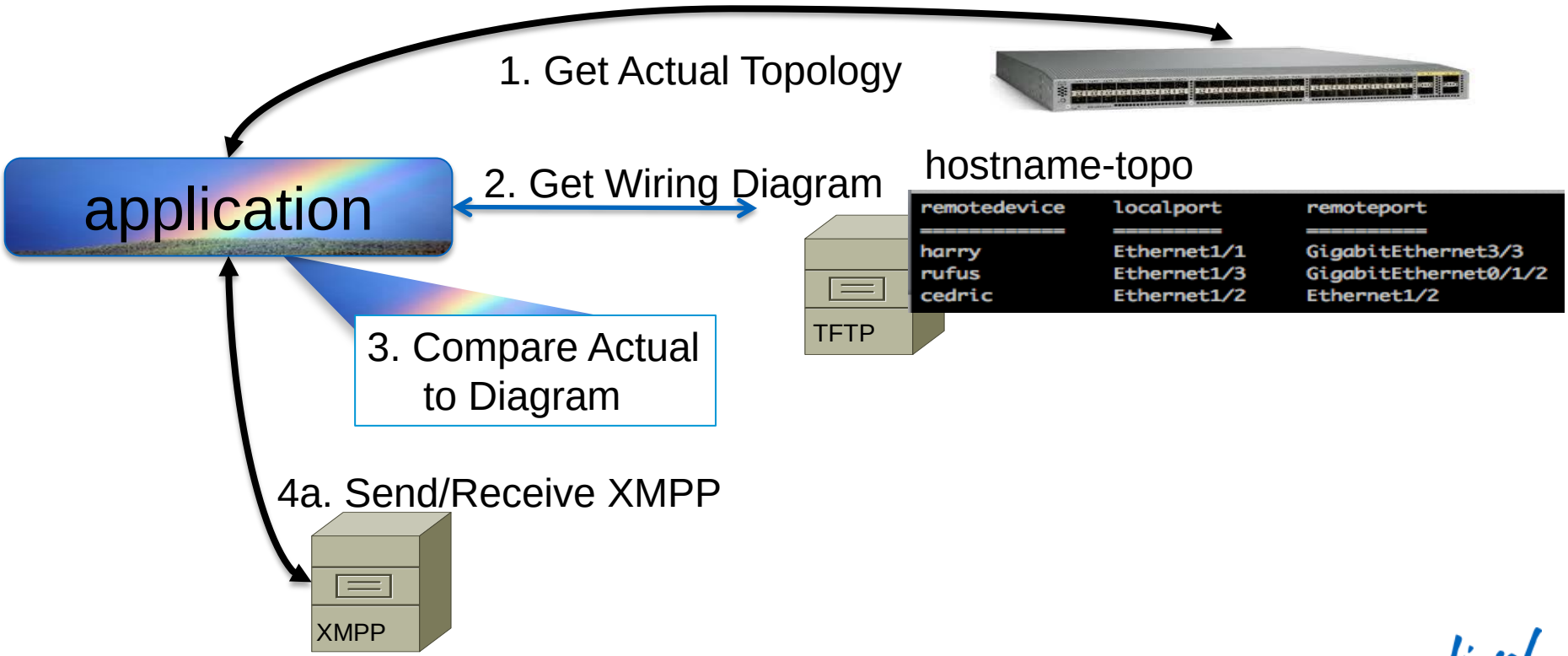
You Had Programmatic Access to Topology Data?



Simplification: Neighbour Data Use Case

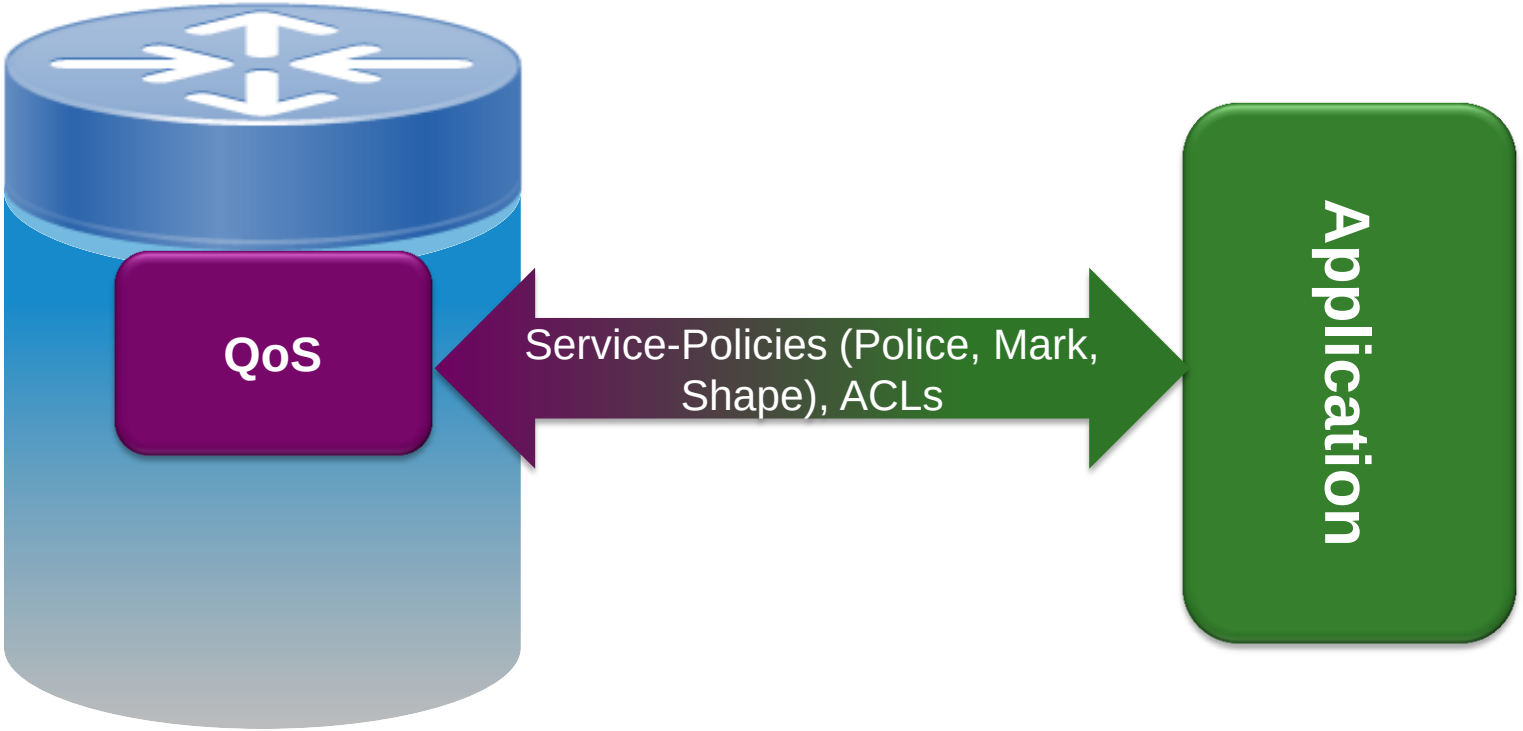


Wiring Verification: Could It Be True?



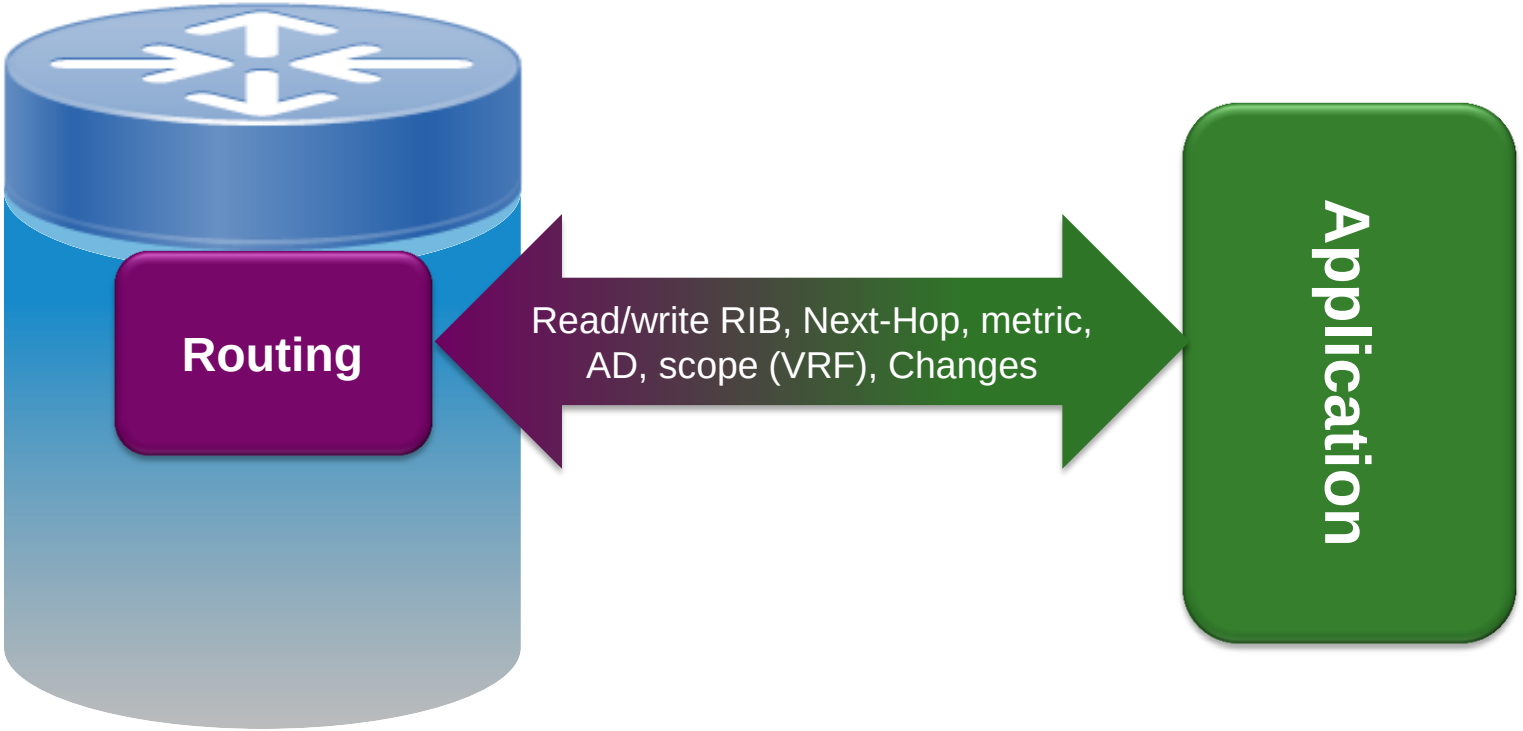
Demo Time

What Would You Do If…You Could Program QoS?



What Would You Do If…

You Could Interact with the RIB?



Getting and Setting Routes

```
L3UnicastScope scope = new L3UnicastScope("", AFIType.IPV4, SAFIType.UNICAST, "");
NetworkPrefix prefix = new NetworkPrefix(InetAddress.getByName("0.0.0.0"), 0);
L3UnicastRIBFilter ribFilter = new L3UnicastRIBFilter(OwnerType.NONE, "NONE", prefix);
L3UnicastRouteRange range = new L3UnicastRouteRange(prefix, RouteRange.RangeType.EQUAL_OR_LARGER, 100);
List<TopoNode> mynodes = TopoNode.getAllNodes();
for(TopoNode thisnode : mynodes) {
    Routing routing = Routing.getInstance(thisnode.ne);
    RIB rib = routing.getRib();
    List<Route> routeList = rib.getRouteList(scope, ribFilter, range);
    for (Route route : routeList) {
```

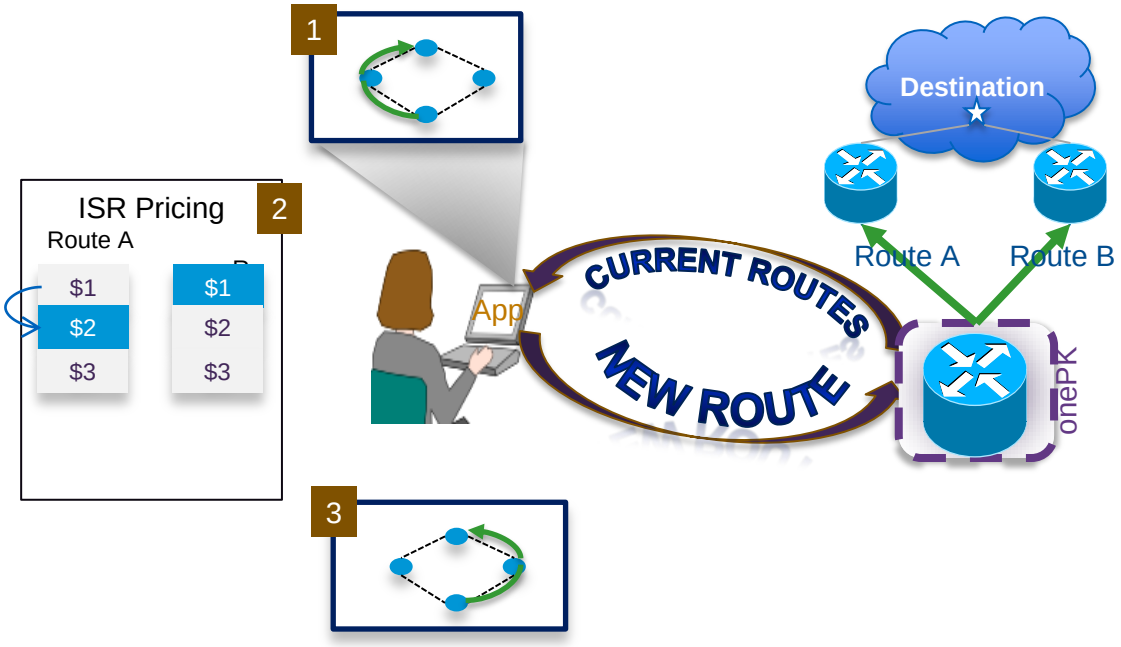
Get Routes

Sample Code.
Subject to Change

```
L3UnicastRoute aRoute = new L3UnicastRoute(prefix, nextHopL
aRoute.setAdminDistance(1);
RouteOperation op = new L3UnicastRouteOperation(RouteOperationType.ADD, aRoute);
List<RouteOperation> opList = new ArrayList<RouteOperation>();
opList.add(op);
AppRouteTable art = routing.getAppRouteTable();
art.updateRoutes(scope, opList);
```

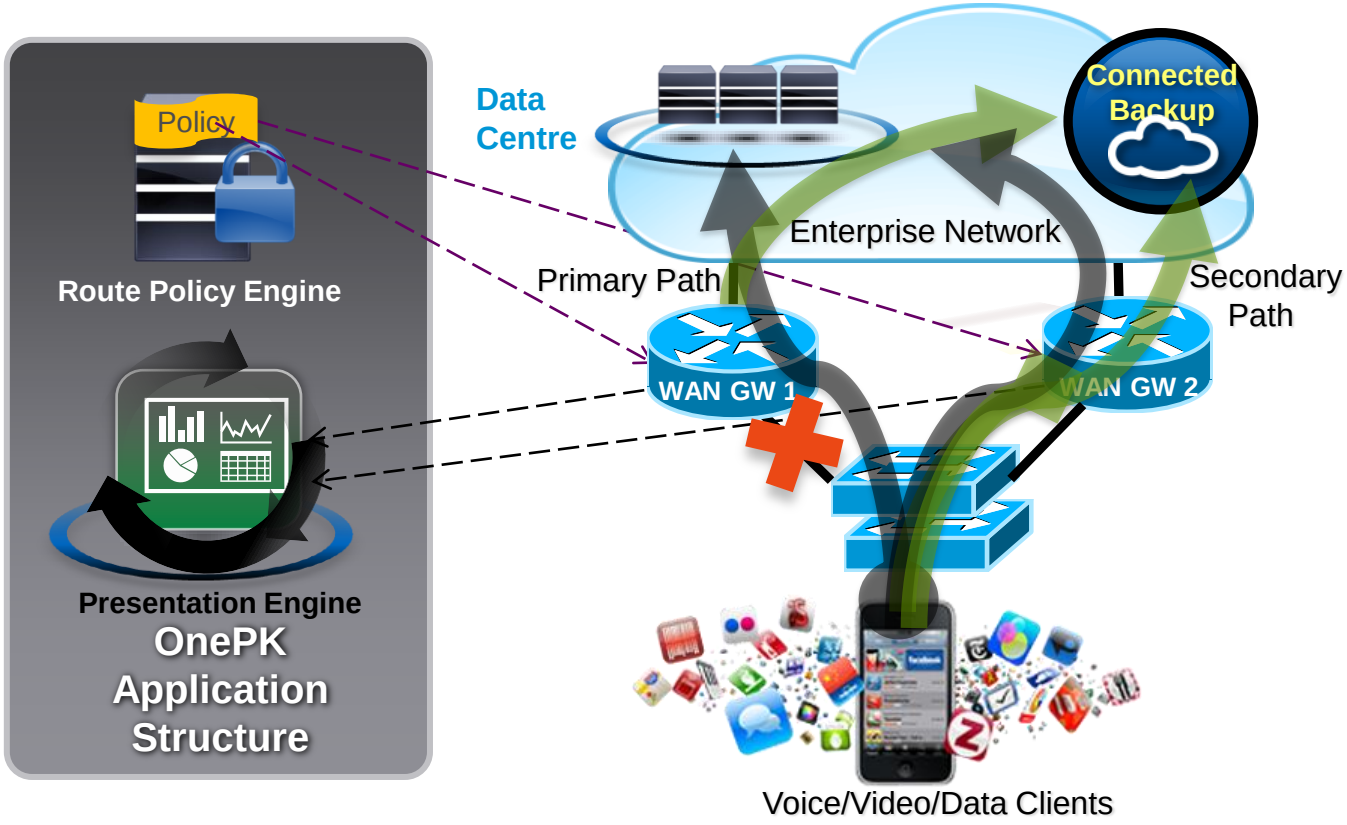
Set Routes

Routing Use Case: Routing For Dollars



Unique Data Forwarding Algorithm Highly Optimised for the Network Operator's Application

Use Case: Traffic Steering for Branch Offices

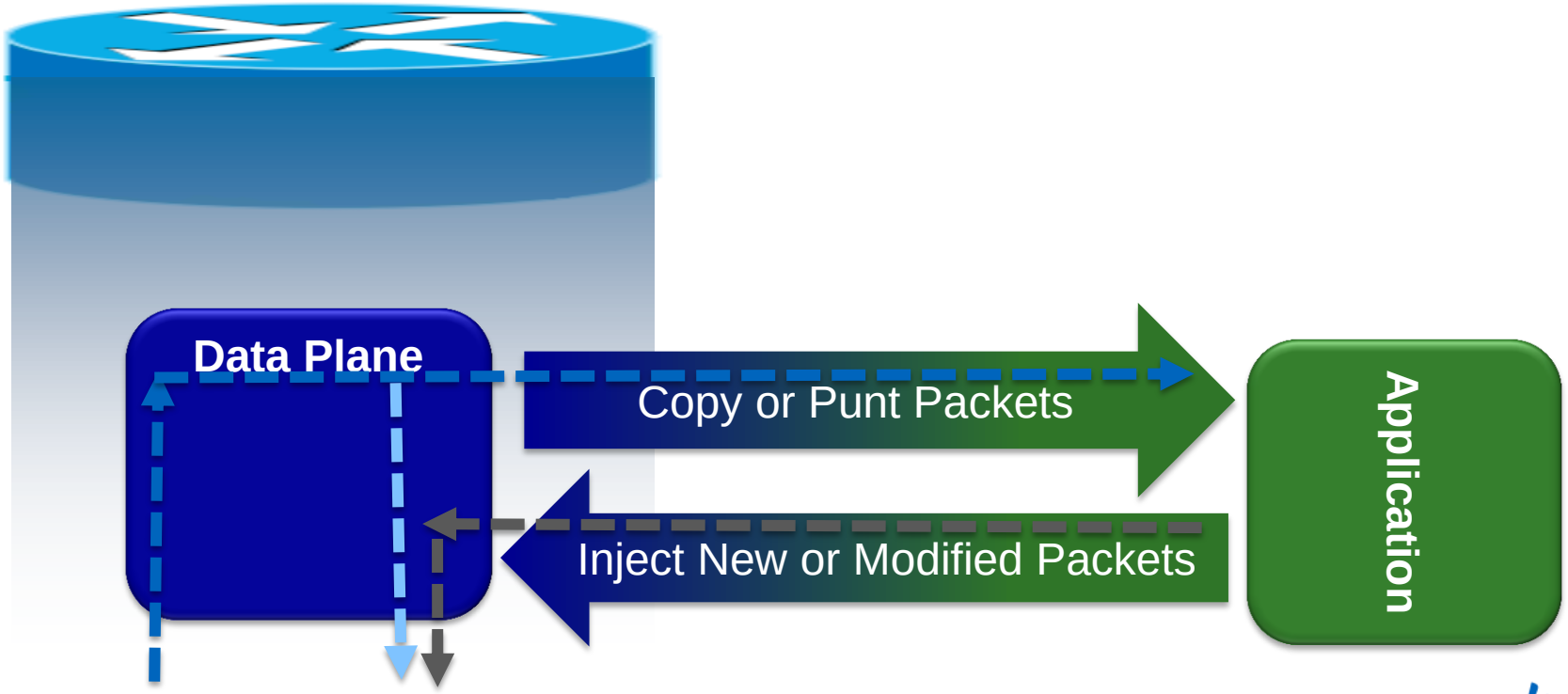


Demo Time

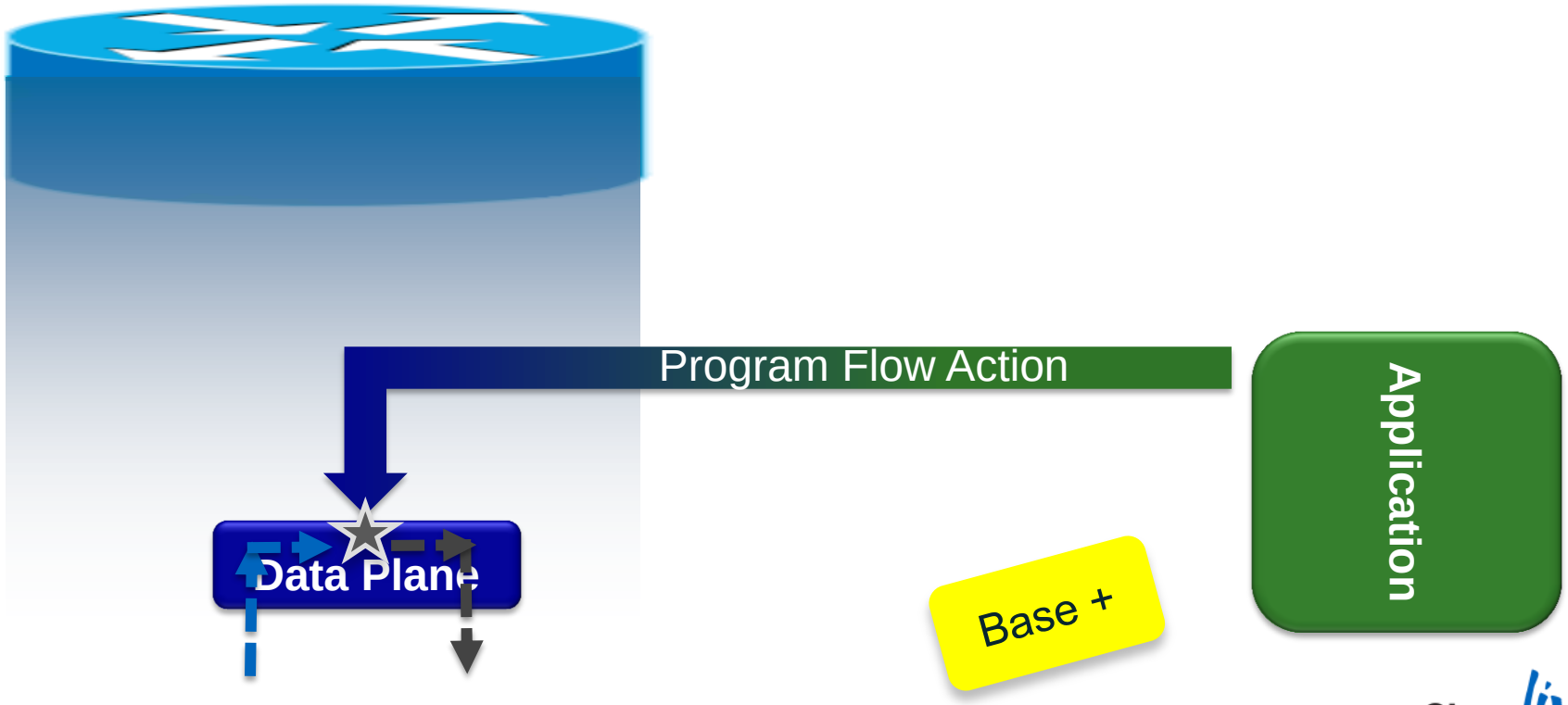
What Could You Do If…

You Could Copy/Punt/Inject Packets?

Data Path Service Set (DPSS) Gives You Hooks Into the Forwarding Plane



Programming Flow Actions



Punting and Injecting Packets

```
TRY(rc, onep_dpss_register_for_packets(
        ne1,
        dpss,
        targ_left,
        interesting_class,
        ONEP_DPSS_ACTION_PUNT,
        encrypt_callback,
        (void *)intf_left,
        &reg_handle), "Register for packets");
```

Defines traffic of interest

Action to take on interesting traffic

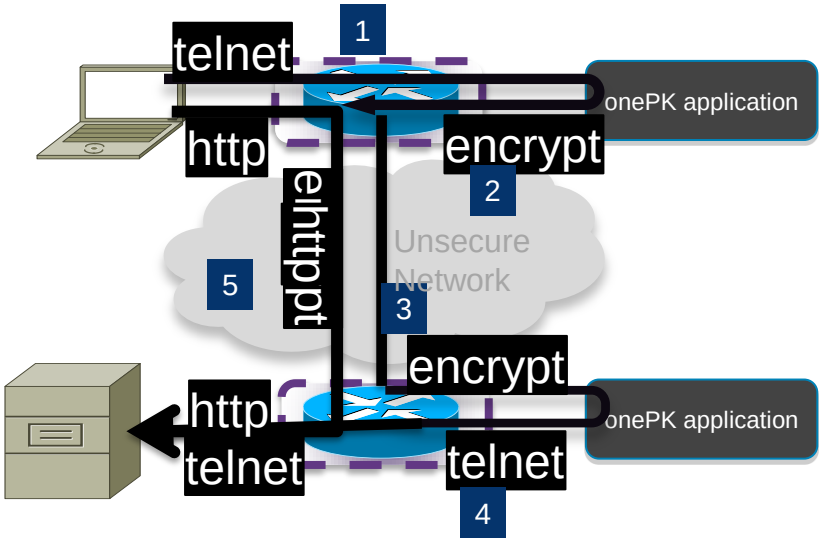
Where traffic goes next

Sample Code Subject to Change

Data Path Use Case: Custom Encryption

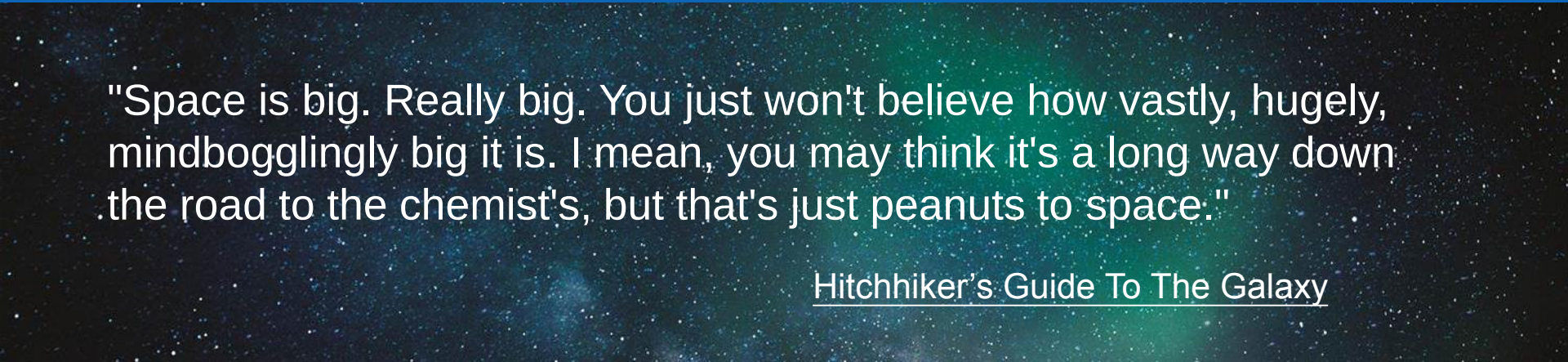
Real Time Modification of Select Traffic

1. Ingress application instructs ingress router to punt telnet and syslog to app
2. Ingress application encrypts punted traffic and re-injects into data path.
3. Egress applications instructs egress router to punt telnet and syslog to app
4. Egress application decrypts punted traffic and re-injects into data path.
5. Traffic that does not match policy passes through unencrypted.



DPSS Demo

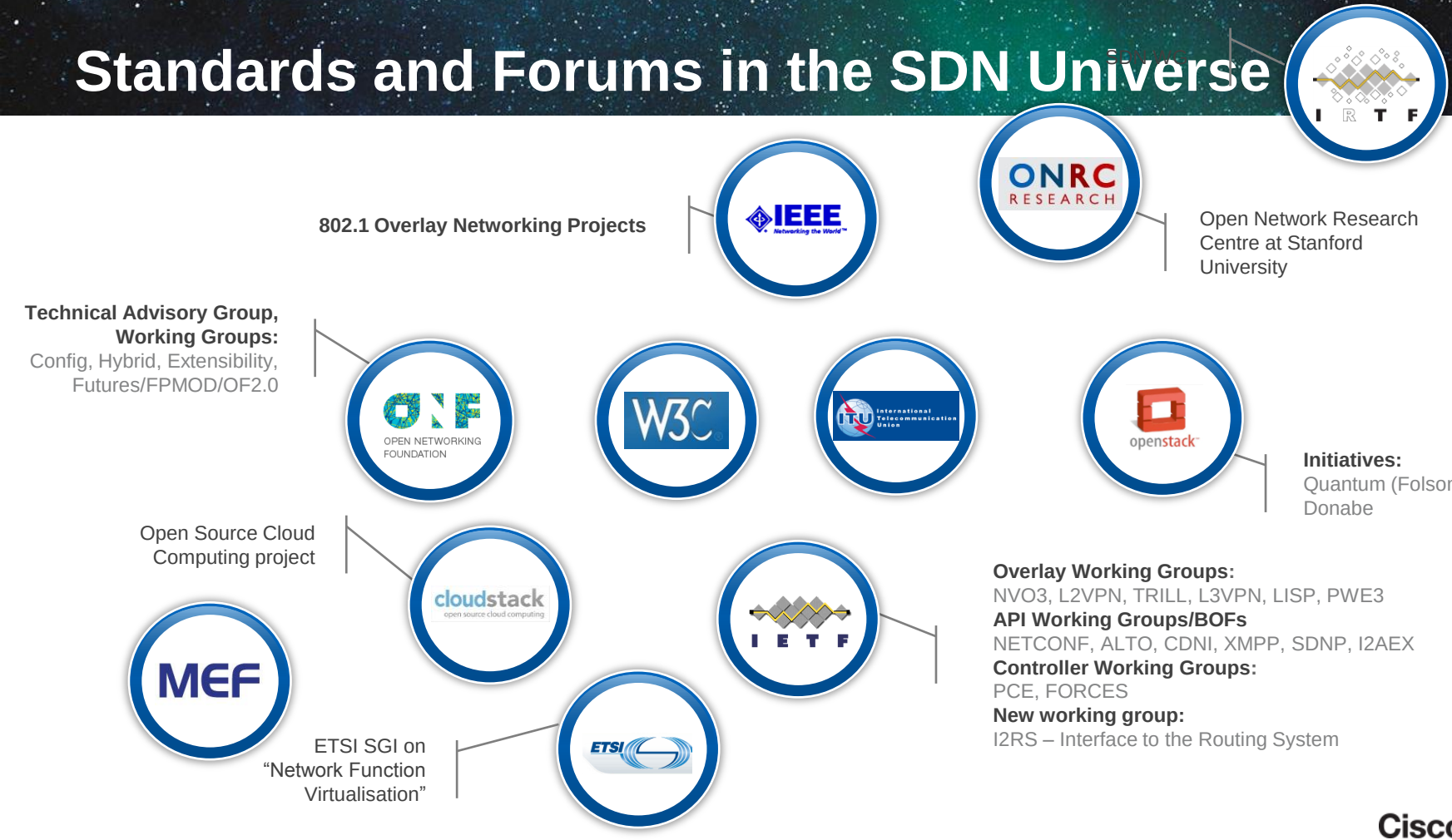
onePK in the SDN Universe



"Space is big. Really big. You just won't believe how vastly, hugely, mindbogglingly big it is. I mean, you may think it's a long way down the road to the chemist's, but that's just peanuts to space."

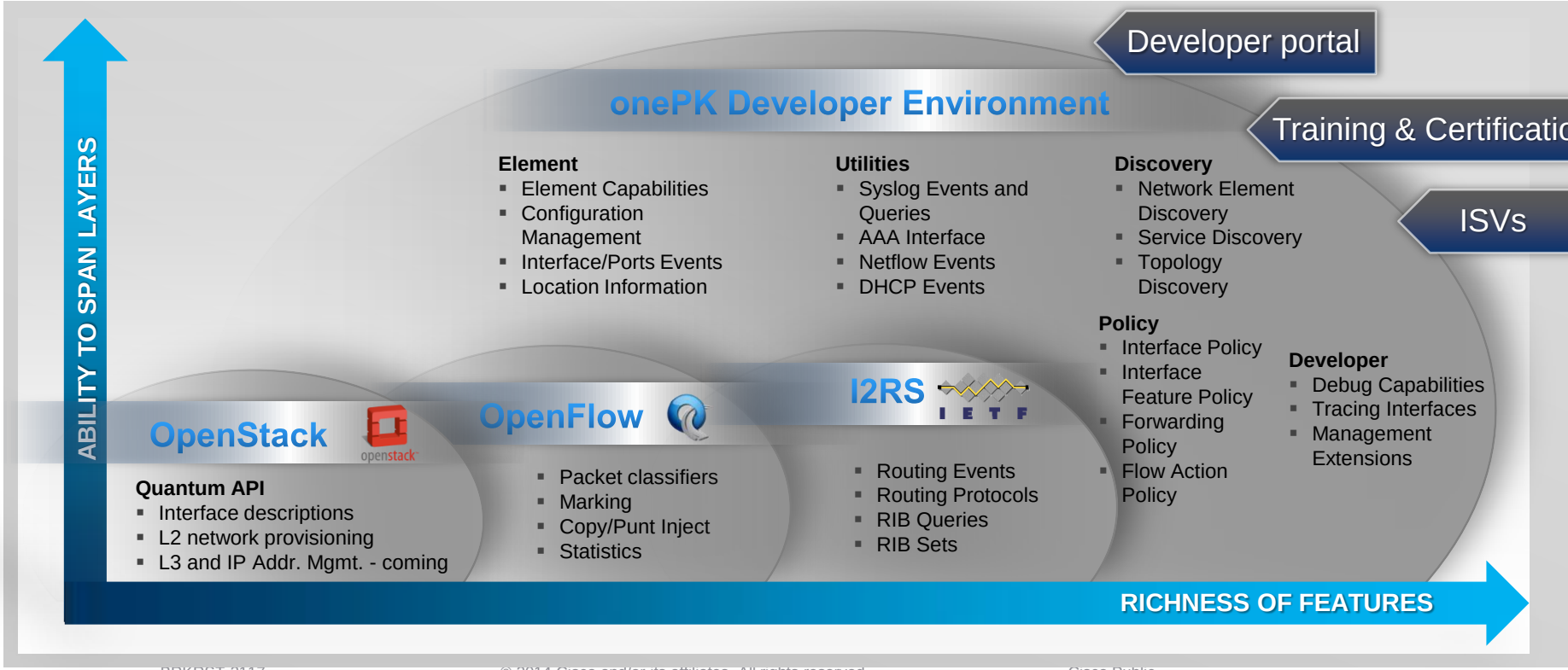
Hitchhiker's Guide To The Galaxy

Standards and Forums in the SDN Universe

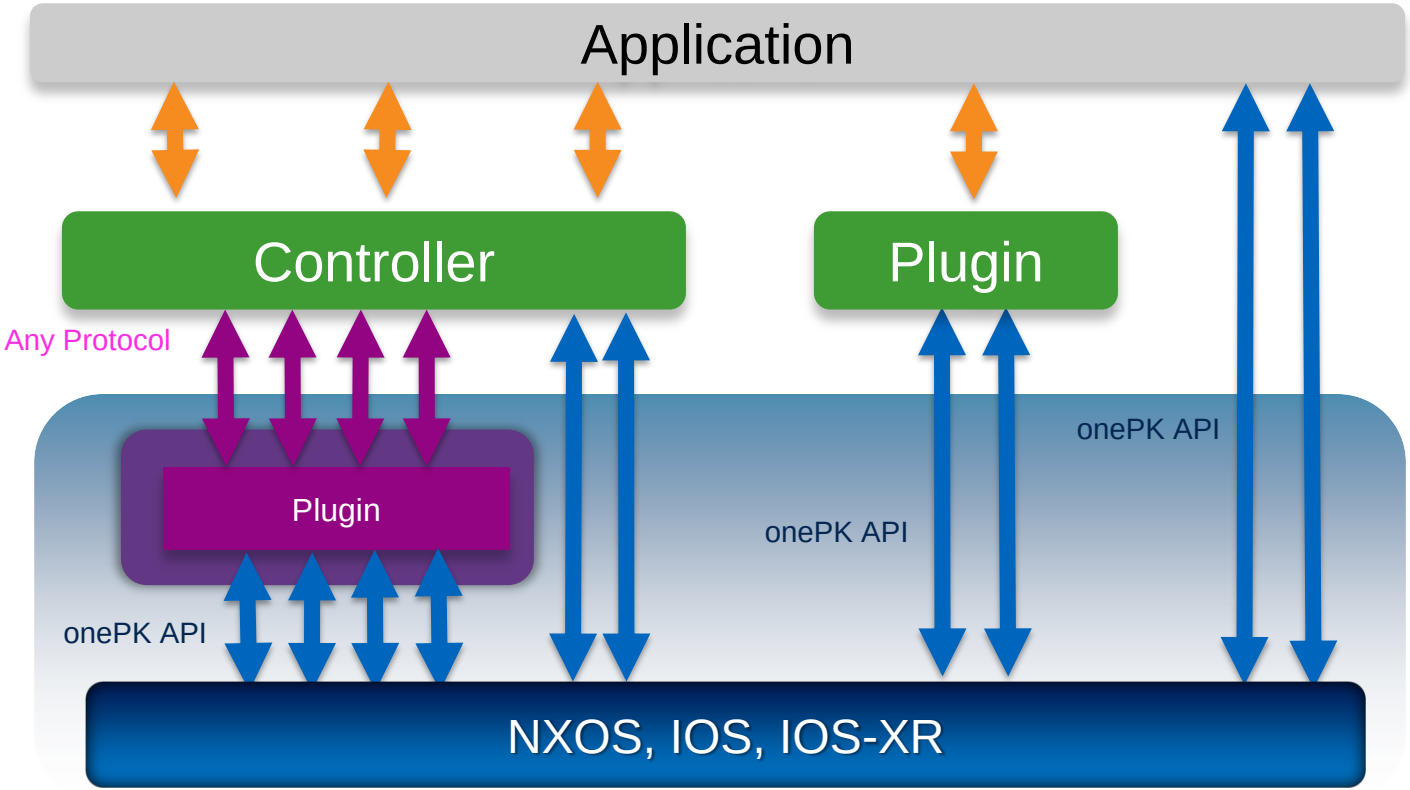


onePK is not Openstack, Openflow, I2RS, …

But We Can All Be Friends

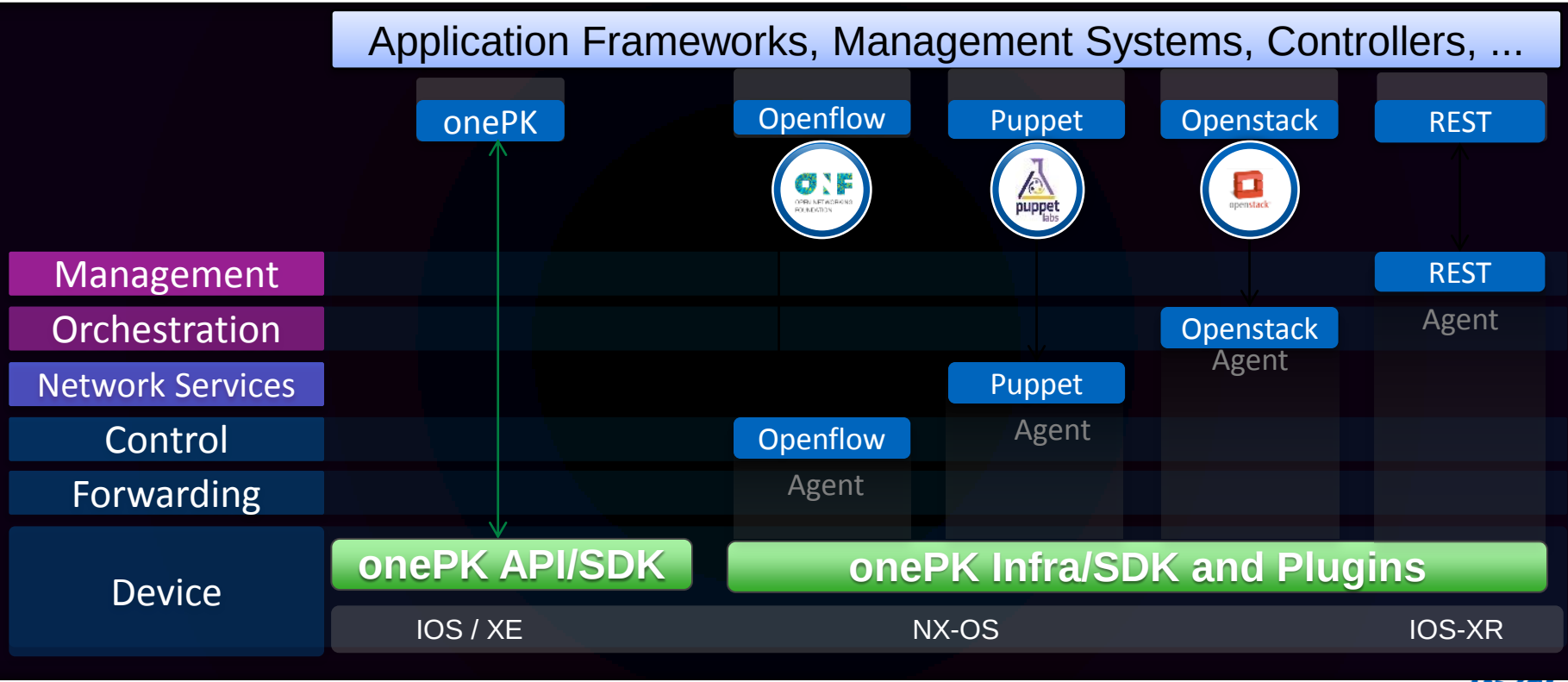


Plugins? Controllers? – Let the Use Case Decide



onePK is a Toolkit

Flexible, Programmatic Network Access






Conclusion

How Network Admins Can Get Ready for onePK

Evaluate Specific Use Cases and Gather Requirements – is onePK the right solution? Do You Need to Go Up the Stack?

Learn the CLI (Provision, Monitor, Maintain, Troubleshoot)

Brush Up On Scripting Skills (Python)

Sass* Some Hoopy* Software Developers

Hire Some Software Developer Froods*

*Sass: Know, Be Aware Of, Meet
*Hoopy: Really Together Guy
*Frood: Really Amazingly Together Guy
--Hitchhiker's Guide to the Galaxy

For All Other Questions, the Answer is…

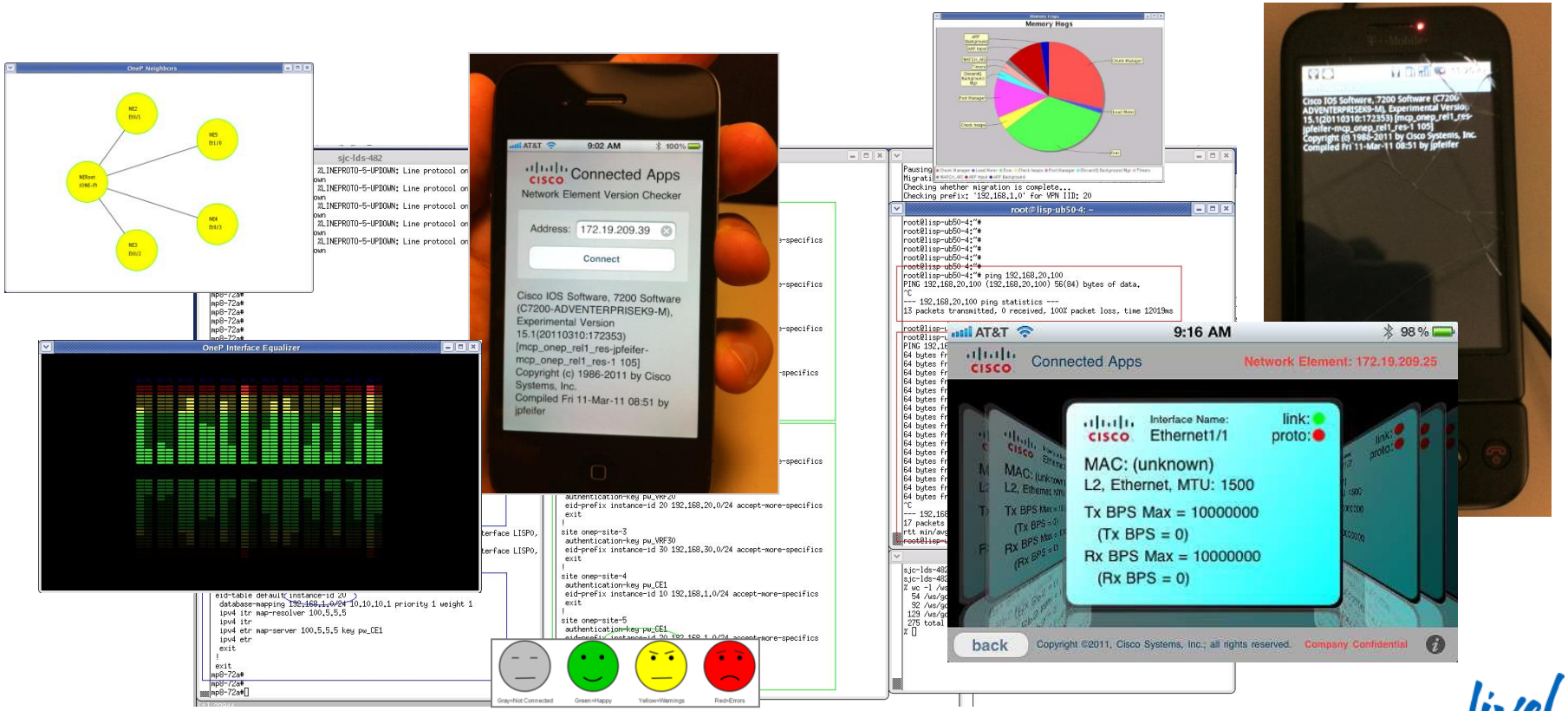
For More Information:

- BRKRST-2051 - Software Defined Networks and OpenFlow
- BRKDCT-2314 - Agile Automated Data-Centers
- LABWISP-1000 - Walk-in Self Paced Labs
- BRKARC-2012 - Application Hosting and OnePK Architecture Overview
- BRKCDN-1005 - Better Network Management Through Network Programmability
- BRKCDN-1015 - onePK: Designing Real World Applications
- BRKCRS-3090 - Implementing Network Automation
- PSODCT-1006 - Cisco ONE: Software Defined Networking Evolution

…and more!

http://developer.cisco.com/web/onepk/home

What Could You Do With onePK?






Q & A

Complete Your Online Session Evaluation

Give us your feedback and receive a Cisco Live 2014 Polo Shirt!

Complete your Overall Event Survey and 5 Session Evaluations.

- Directly from your mobile device on the Cisco Live Mobile App
- By visiting the Cisco Live Mobile Site www.ciscoliveaustralia.com/mobile
- Visit any Cisco Live Internet Station located throughout the venue

Polo Shirts can be collected in the World of Solutions on Friday 21 March 12:00pm - 2:00pm

Learn online with Cisco Live!

Visit us online after the conference for full access to session videos and presentations.
www.CiscoLiveAPAC.com


CISCO
TM