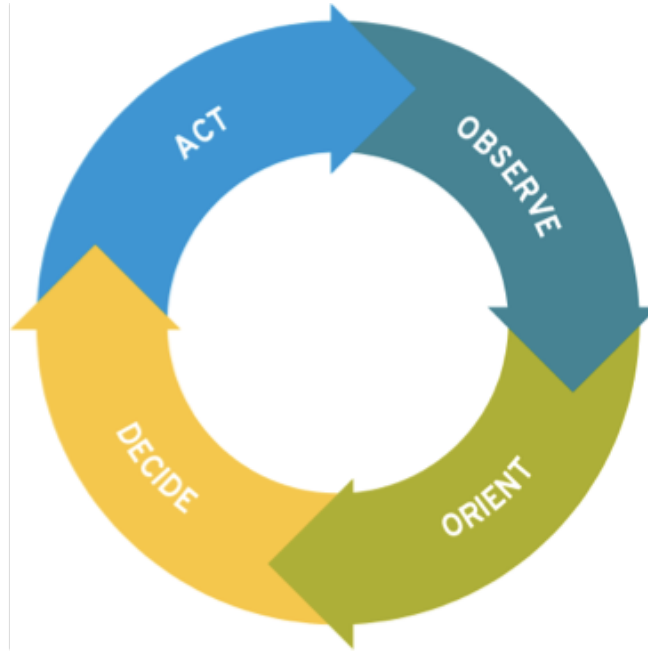


Scalable Fabric Playbook for IXPs : Automation Options for Interconnecting Internet, Content and Cloud

Mikael Holmberg
Distinguished Systems Engineer



Manual operations
Custom scripts
....

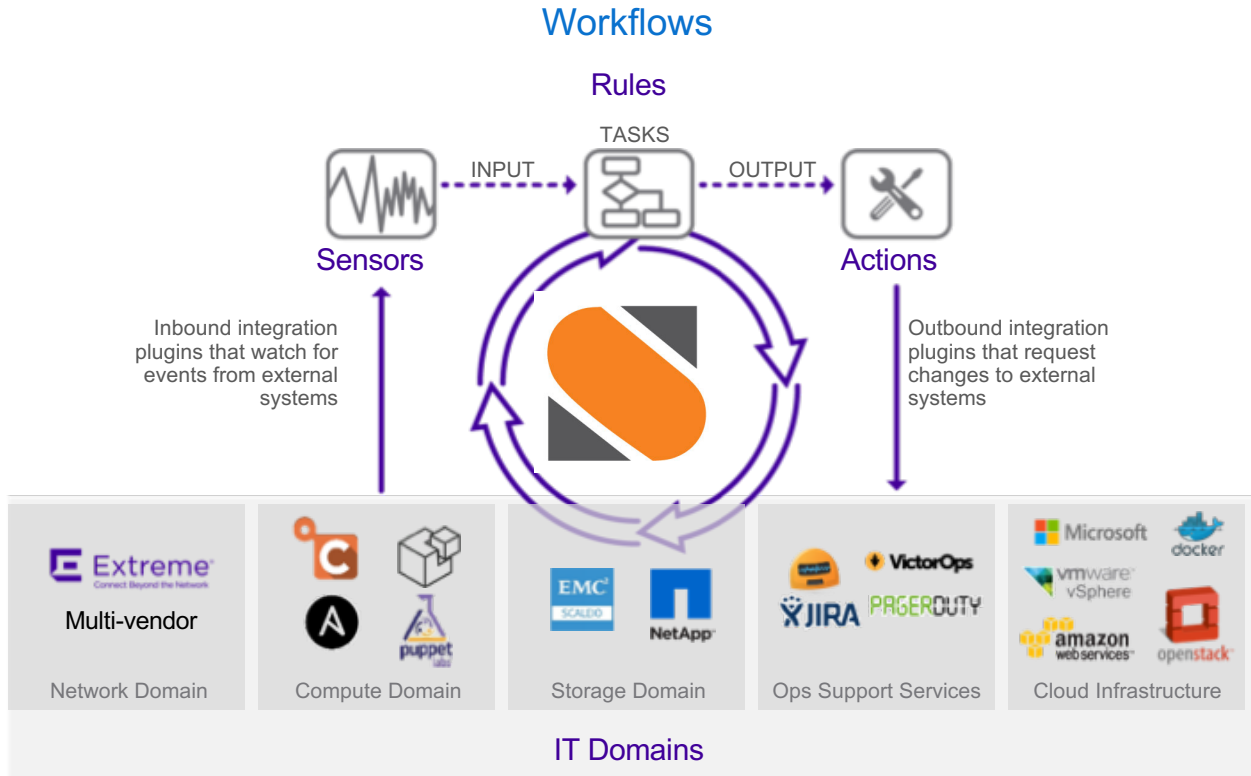


Event Driven Automation



OpenSource Event Automation Platform

Event Driven Automation



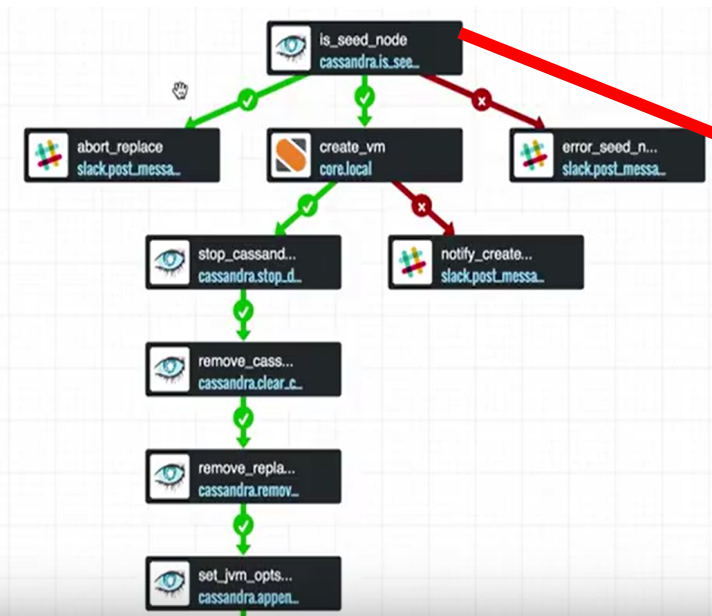
Sensors : Listen for Events like outage in an area

Actions: How to make the change via tools or Stacktorm

Workflow anatomy

Workflow : Cassandra.replace_host

Tasks



Workflow

Rules TASKS

INPUT OUTPUT

Sensors

Actions

```
version: '2.0'

cassandra.replace_host:
  description: A basic workflow that replaces a dead cassandra node with a spare.
  type: direct
  input:
    - dead_node
    - replacement_node
    - healthy_node
  output:
    - just_output_the_whole_workflow_context: "<% $ %%"
  tasks:
    is_seed_node:
      action: cassandra.is_seed_node
      input:
        hosts: "<% $.healthy_node %%"
        node_id: "<% $.dead_node %%"
      publish:
        seed_node: "<% $.is_seed_node.get($.healthy_node).stdout %%"
      on-success:
        - abort_replace: "<% $.seed_node = 'True' %%"
        - create_vm: "<% $.seed_node = 'False' %%"
        - error_seed_node_determination: "<% not $.seed_node in list(False, True) %%"
      on-error:
        - error_seed_node_determination
    abort_replace:
      action: slack.post_message
      input:
        channel: "#dsedemo"
        message: "``[CASS-REPLACE-HOST] [<% $.dead_node %%] STATUS: FAILED REASON: SE
      on-complete:
        - fail
    error_seed_node_determination:
      action: slack.post_message
      input:
```



Workflow representation and code side by side

Auto-remediating Cassandra with StackStorm

The screenshot displays a web browser window with the URL `st2-dse-demo-controller001.uswest2.stackstorm.net:8080/flow/#/action/cassandra.replace_host`. The interface is split into three main sections:

- Left Panel:** A sidebar with a search bar and a list of AWS actions, including `aws.create_vm`, `aws.destroy_vm`, `aws.ec2_allocate_address`, `aws.ec2_assign_private_ip_ad_`, `aws.ec2_associate_address`, `aws.ec2_associate_address_ob_`, `aws.ec2_attach_network_interf_`, and `aws.ec2_attach_volume`.
- Center Panel:** A visual workflow diagram. It starts with a task `is_seed_node` (cassandra.is_see...). A green arrow leads to `create_vm` (core.local), which then leads to `stop_cassand...` (cassandra.stop_d...), `remove_cass...` (cassandra.clear_c...), `remove_repla...` (cassandra.remov...), and finally `set_jvm_opts...` (cassandra.appen...). A red arrow from the start task leads to `abort_replace` (slack.post_messa...). A red 'X' mark is visible on the right side of the diagram.
- Right Panel:** A code editor showing the workflow's configuration in JSON format. The code includes a version, a description, input/output fields, and a series of tasks and actions.

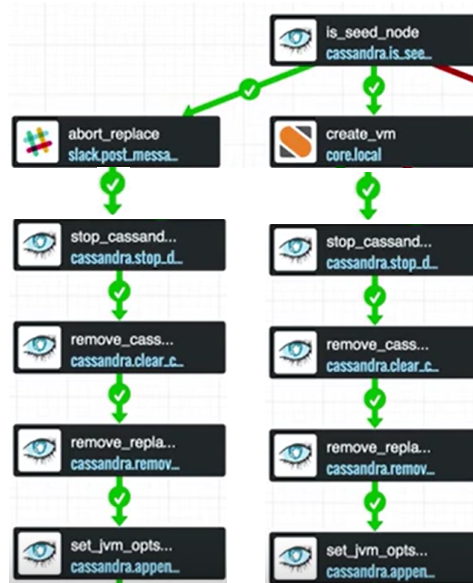
```
1 version: '2.0'
2
3 cassandra.replace_host:
4   description: A basic workflow that replaces a c
5   type: direct
6   input:
7     - dead_node
8     - replacement_node
9     - healthy_node
10  output:
11    just_output_the_whole_workflow_context: "<
12  tasks:
13    is_seed_node:
14      action: cassandra.is_seed_node
15      input:
16        hosts: "<
17        node_id: "<
18      publish:
19        seed_node: "<
20    on-success:
21      - abort_replace: "<
22      - create_vm: "<
23      - error_seed_node_determination: "<
24    on-error:
25      - error_seed_node_determination
26  abort_replace:
27    action: slack.post_message
28    input:
29      channel: "#dse-demo"
30      message: ""[CASS-REPLACE-HOST] [<
31  on-complete:
32    - fail
```

Tasks in Workflow: Linear , semi parallel or parallel

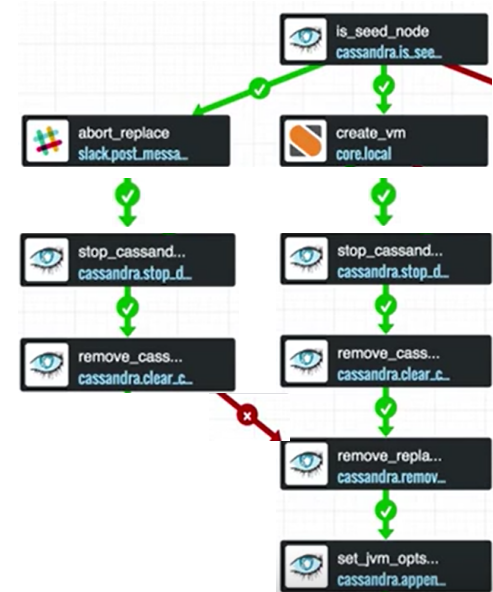
Linear



Parallel

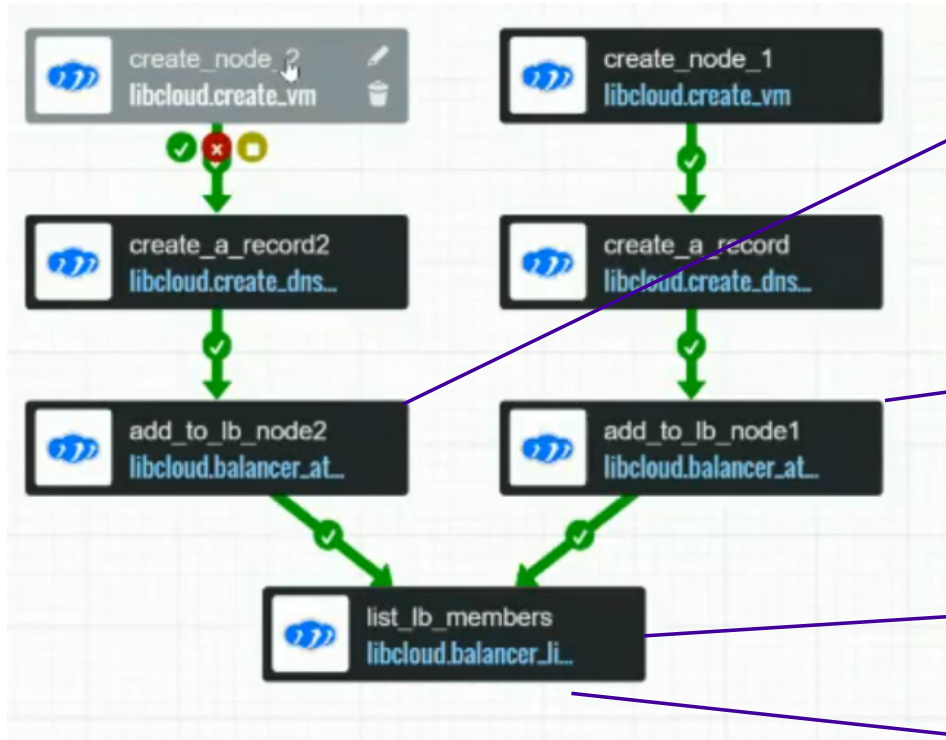


Semi-parallel



Tasks can collapse with “Joins”

tasks:



add_to_lb_node2

action:

input:

publish:...



on-success:

- list_lb_members

- notify

add_to_lb_node1

action:

input:

publish:...



on-success:

- list_lb_members

- notify

list_lb_members

join: all

input: ...

publish:...

on-success:

-close_request

- notify

Retry policies : for example reboot vm and wait for node to reboot

```
9  workflows:
10
11      main:
12          type: direct
13          tasks:
14              init:
15                  action: core.local cmd="rm -f /tmp/done"
16                  on-success:
17                      - create-file
18                      - test-error-undo-retry
19              create-file:
20                  action: core.local cmd="touch /tmp/done"
21                  wait-before: 10
22              test-error-undo-retry:
23                  workflow: work
24              retry:
25                  count: 30
26                  delay: 1
27                  on-success:
28                      - delete-file
29              delete-file:
30                  action: core.local cmd="rm -f /tmp/done"
```


If you like writing visual code.....

The screenshot displays the StackStorm Flow interface. On the left is a sidebar with a search bar and a list of integrations: AWS, Consul, ConsulGet (Get value from Consul server), ConsulParseNodes (Helper function to extract hosts from a Consul response to be used with StackStorm workflows), ConsulPut (Put value in Consul server), ConsulQueryNode (Query details about a node in Consul), ConsulQueryService (Query details about a service in Consul), Core, and CoreHttp (Action that performs an http request). The main workspace shows a workflow diagram for 'st2-demos.diskspace_remediation'. The workflow starts with a 'silence_check' task (sensu.silence), which leads to a 'check_dir_size' task (st2-demos.check_dir_size). From 'check_dir_size', the flow branches: one path goes to 'remove_files' (core.remove_files) and then to 'validate_dir' (st2-demos.validate_dir), which finally leads to 'post_success' (slack.post_message). Another path from 'check_dir_size' goes to 'victorops_esca...' (victorops.open_incident). A third path goes from 'silence_check' to 'victorops_esca...'. The right panel shows the YAML code for the workflow, with the 'check_dir_size' task highlighted in blue.

```
1 ---
2 version: '2.0'
3 name: st2-demos.diskspace_remediation
4
5 workflows:
6   main:
7     input:
8       - hostname
9       - directory
10      - file_extension
11      - threshold
12      - event_id
13      - check_name
14      - alert_message
15      - raw_payload
16
17 tasks:
18   silence_check:
19     # [215, 26]
20     action: sensu.silence
21     input:
22       client: <X $.hostname %>
23       check: <X $.check_name %>
24     on-success:
25       - check_dir_size
26     on-error:
27       - victorops_escalation
28
29   check_dir_size:
30     action: st2-demos.check_dir_size
31     input:
32       hosts: <X $.hostname %>
33       directory: <X $.directory %>
34       threshold: <X $.threshold %>
35     on-error:
36       - remove_files
37     on-success:
38       - victorops_escalation
39
40   remove_files:
41     # [355, 230]
42     action: core.remove_files
```

Select the workflow

Click on the task you want to edit

Write only that portion of the highlighted code

Stackstorm Integration Packs

Integration packs:

<https://exchange.stackstorm.org/>

Cloud Providers



aws



azure



dimensiondata



libcloud



rackspace

Automations and Monitoring



EXOS



Network Essentials



servicenow



datadog



sensu



newrelic



mmonit



icinga2



dripstat

Essentials



ansible



napalm



slack



chef



splunk



cloudflare



email



elasticsearch



docker



excel

Curiosities



astral



cubesensors



hue



nest



powerpoint



urbandict



save_kittens



tesla



Working with Integration Packs

Managing Packs

List all installed packs

```
st2 pack list
```

Get detailed information about an installed pack

```
st2 pack get core
```

Discovering Packs

Search query is applied across all pack parameters.

It will search through pack names:

```
st2 pack search sensu
```

And keywords:

```
st2 pack search monitoring
```

And description (use quotes for multi-word search):

```
st2 pack search "Amazon Web Services"
```

And even pack author:

```
st2 pack search "Jon Middleton"
```

Show an index entry for the pack

with the exact name match

```
st2 pack show sensu
```

Installing a Pack

Fetch a specific commit

```
st2 pack install cloudflare=776b9a4
```

Or a version tag

```
st2 pack install cloudflare=0.1.0
```

Or a branch

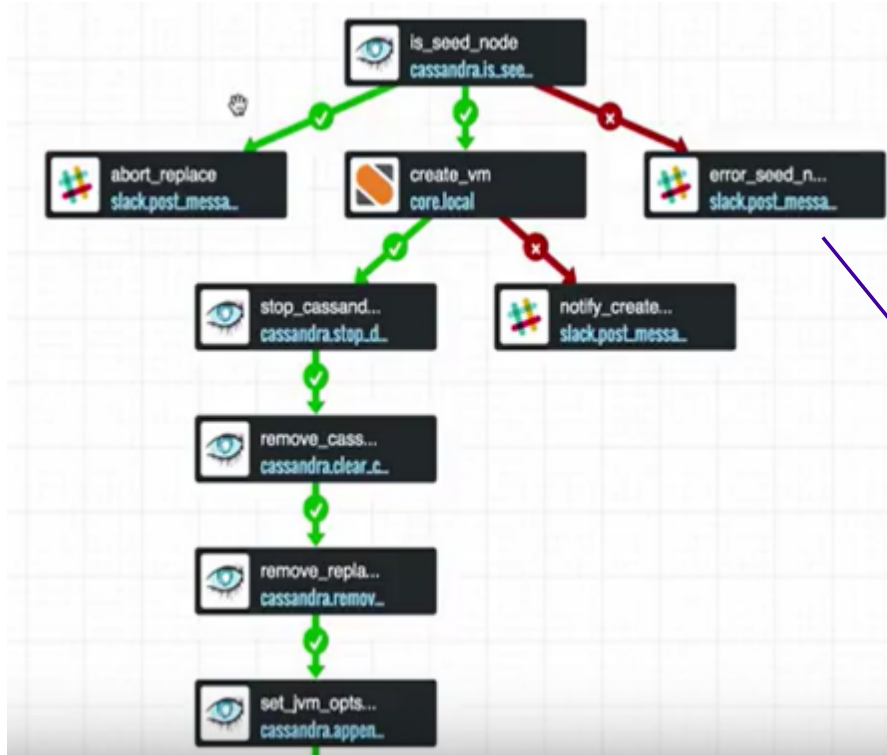
```
st2 pack install https://github.com/emedvedev/chatops_tutorial=testing
```

Configuring a Pack

```
st2 pack config cloudflare
```



Example #1 ChatOps Pack : Notify others on job status



cassandra.replace.host:
type:direct
input:...
output:...
tasks:

is_seed_node:

action:
input:
publish:...
on-success:
- **error_seed_notify:**
- **create_vm:**

on-error:
- error_seed_node_determination:...

error_seed_notify:

action: slack.post.message
input:
channel: "#IXPOperations"
message: "...Error: Cassandra replace host..."

Example #2 Excel Pack : Load Information in workflow

The screenshot shows a workflow editor interface. On the left, a sidebar lists various packs: chatops, core, dcfabric, default, and excel. The 'excel' pack is highlighted with a red box. Below the packs, a list of tasks is shown: get_keys_for_columns, get_keys_for_rows, get_sheets, get_variables, and set_variables. The 'get_variables' task is highlighted with a red box. In the center workspace, a task card for 'get_variables' is placed, with the label 'excel.get_variables' below it. A purple arrow points from the 'get_variables' task in the sidebar to the task card in the workspace. Another purple arrow points from the task card to a code editor on the right. The code editor shows a workflow definition with the following content:

```
1 ---  
2 version: '2.0'  
3  
4 * untitled:  
5   tasks:  
6     task1:  
7       # [198, 60]  
8       action: excel.get_variables  
9
```



Create task
Select Excel pack
Choose get_variables
Name the task

Specify Excel file location in Add Metadata

Example #2 Excel Pack : Load Information in workflow

Add Excel Parameters for **inputs**

Excel sheet name

key  

Key to get

New parameter

Name *

variables

Type *

string

Description

Target a specific or subset of variables in JSON format

Enum

Default

☐ required ☐ immutable ☐ secret

```
{
  "name": "create_l2_tenant_demo",
  "pack": "default",
  "runner_type": "mistral-v2",
  "enabled": true,
  "entry_point": "workflows/create_l2_tenant_demo.yaml",
  "parameters": {
    "excel_file": {
      "type": "string",
      "description": "Name of Excel file with path",
      "default": "/opt/stackstorm/packs/excel/rbridge_info.xlsx"
    },
    "key_column": {
      "type": "integer",
      "description": "Starting column for keys",
      "default": 1
    },
    "variable_name_row": {
      "type": "integer",
      "description": "Starting row for variable names",
      "default": 1
    },
    "sheet": {
      "type": "string",
      "description": "Excel sheet name",
      "default": "Fabric"
    },
    "key": {
      "type": "string",
      "description": "Key to get"
    }
  },
  "ref": "default.create_l2_tenant_demo",
  "description": "Add a server to the network"
}
```

Parameters Added

Example #2 Excel Pack : Create Workflow

Search...

- add_ipv4_rule_acl
Add an L3 IPv4 ACL rule to an existing ACL
- add_ipv6_rule_acl
Add an L3 IPv6 ACL rule to an existing ACL
- add_or_remove_l2_acl_rule
Add or remove an ACL rule to or from an L2 ACL
- apply_acl
Apply an ACL to a physical port, port channel, VE or management interface.
- autopick_port_channel_id
Auto pick the lowest available Port Channel ID on the device.
- configure_mac_move_detection
Configure the switch to enable MAC move detection and set the threshold for number of MAC moves in time-window to trigger the detection.
- configure_mgmt_virtual_ip
Configure management virtual IP on the principal switch of a VCS cluster.
- create_acl
Create an Access Control List
- create_l2_port_channel
Create an L2 port channel

get_variables
excel.get_variables

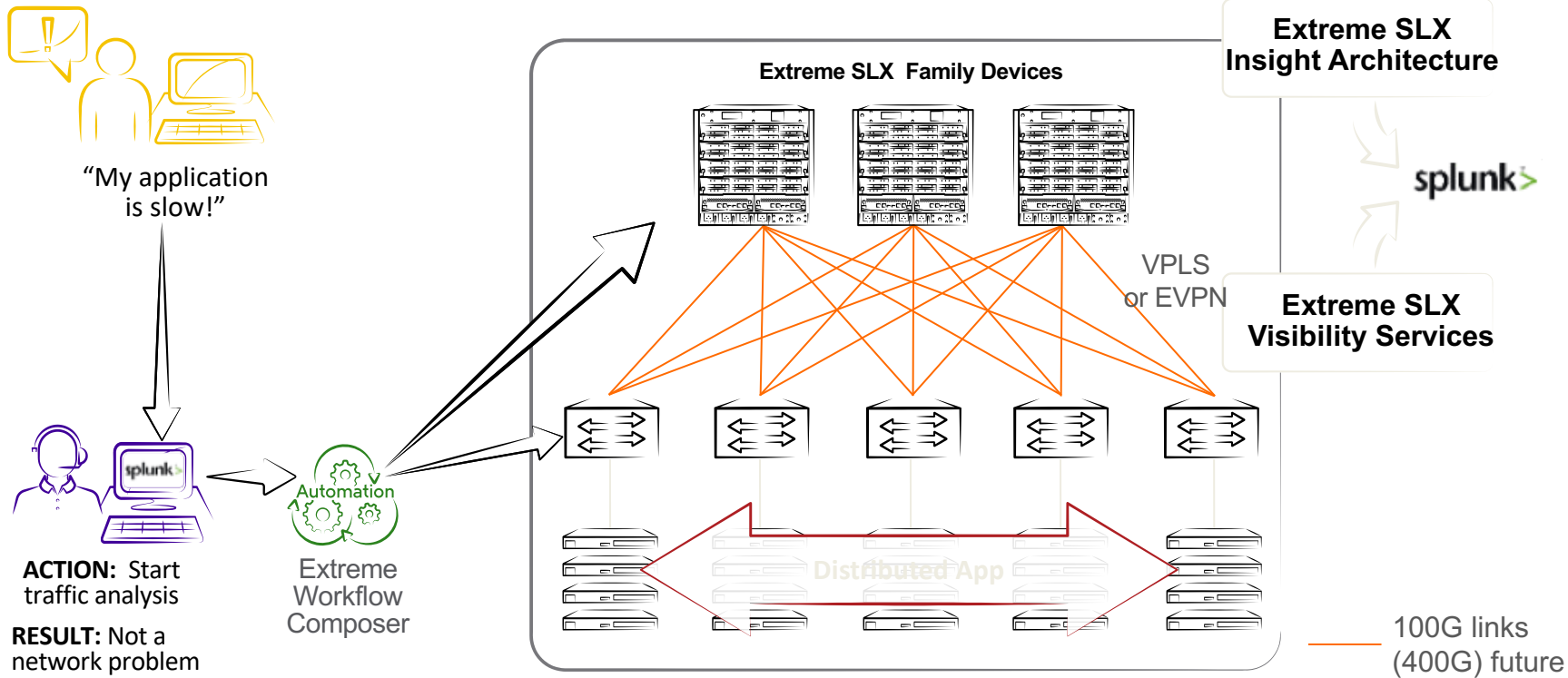
autopick_port_channel_id
network_essentials.autopick_port_channel_id

```
1 ---
2 version: '2.0'
3
4 default.create_l2_tenant_demo:
5   input:
6     - variable_name_row
7     - key_column
8     - variables
9     - excel_file
10    - key
11    - sheet
12    - mgmt_ip
13    - username
14    - password
15   tasks:
16     - get_variables:
17       # [105, 26]
18       action: excel.get_variables
19       input:
20         excel_file: <% $.excel_file %>
21         key_column: <% $.key_column %>
22         variable_name_row: <% $.variable_name_row %>
23         sheet: <% $.sheet %>
24         key: <% $.key %>
25         variables: <% $.variables %>
26     - publish:
27       ports: '<% (task(get_variables).result.result.ports).split(",") %>'
28       mgmt_ip: '<% task(get_variables).result.result.mgmt_ip %>'
29       vlan_id: '<% str(task(get_variables).result.result.vlan_id) %>'
30       intf_type: '<% task(get_variables).result.result.intf_type %>'
31       intf_desc: '<% task(get_variables).result.result.intf_desc %>'
32       protocol: '<% task(get_variables).result.result.protocol %>'
33       mode: '<% task(get_variables).result.result.mode %>'
34     - on-success:
35       - autopick_port_channel
36       autopick_port_channel:
37         # [105, 128]
38         action: network_essentials.autopick_port_channel_id
39         input:
40           mgmt_ip: <% $.mgmt_ip %>
41           username: <% $.username %>
42           password: <% $.password %>
43         publish:
44           port_channel_id: '<% task(autopick_port_channel).result.result.port_channel_id %>'
45
```

Create port channel based on information of ports from Excel file

Agility through Automation and Visibility

Accelerating mean-time-to-innocence through automation for IXPs

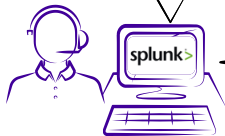


Agility through Automation and Visibility

Accelerating mean-time-to-innocence through automation for IXPs



"My application is slow!"

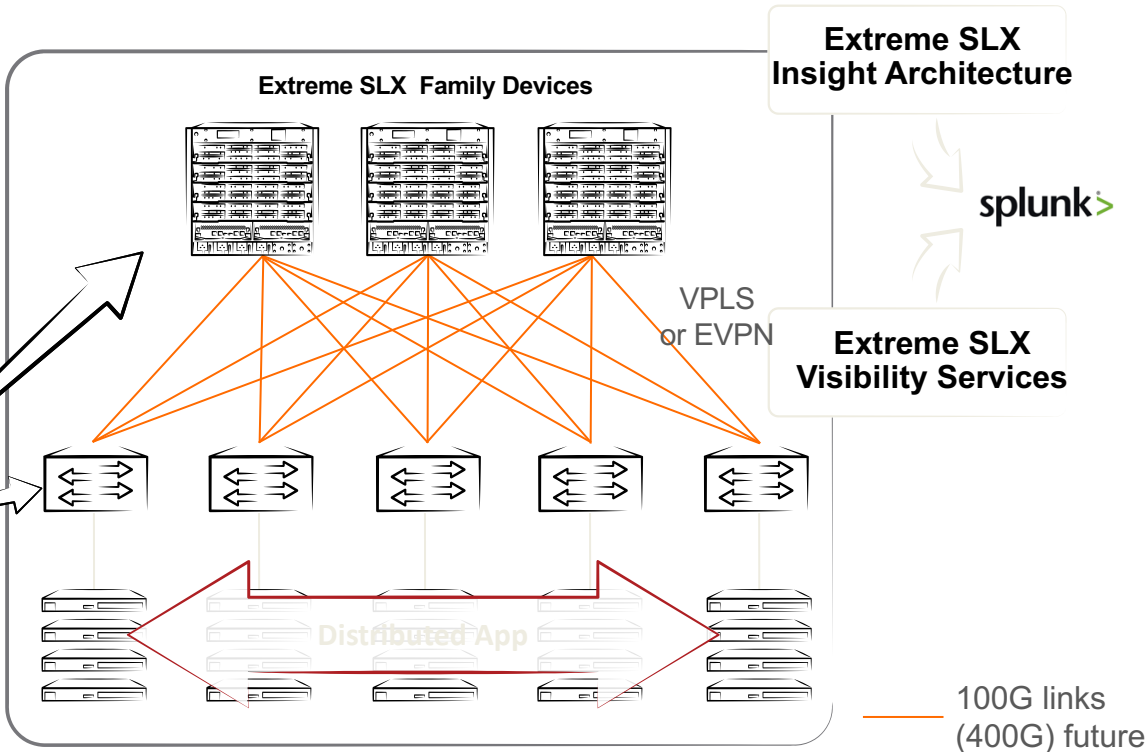


ACTION: Start traffic analysis

RESULT: Not a network problem



Extreme Workflow Composer





Thank You

WWW.EXTREMENETWORKS.COM

