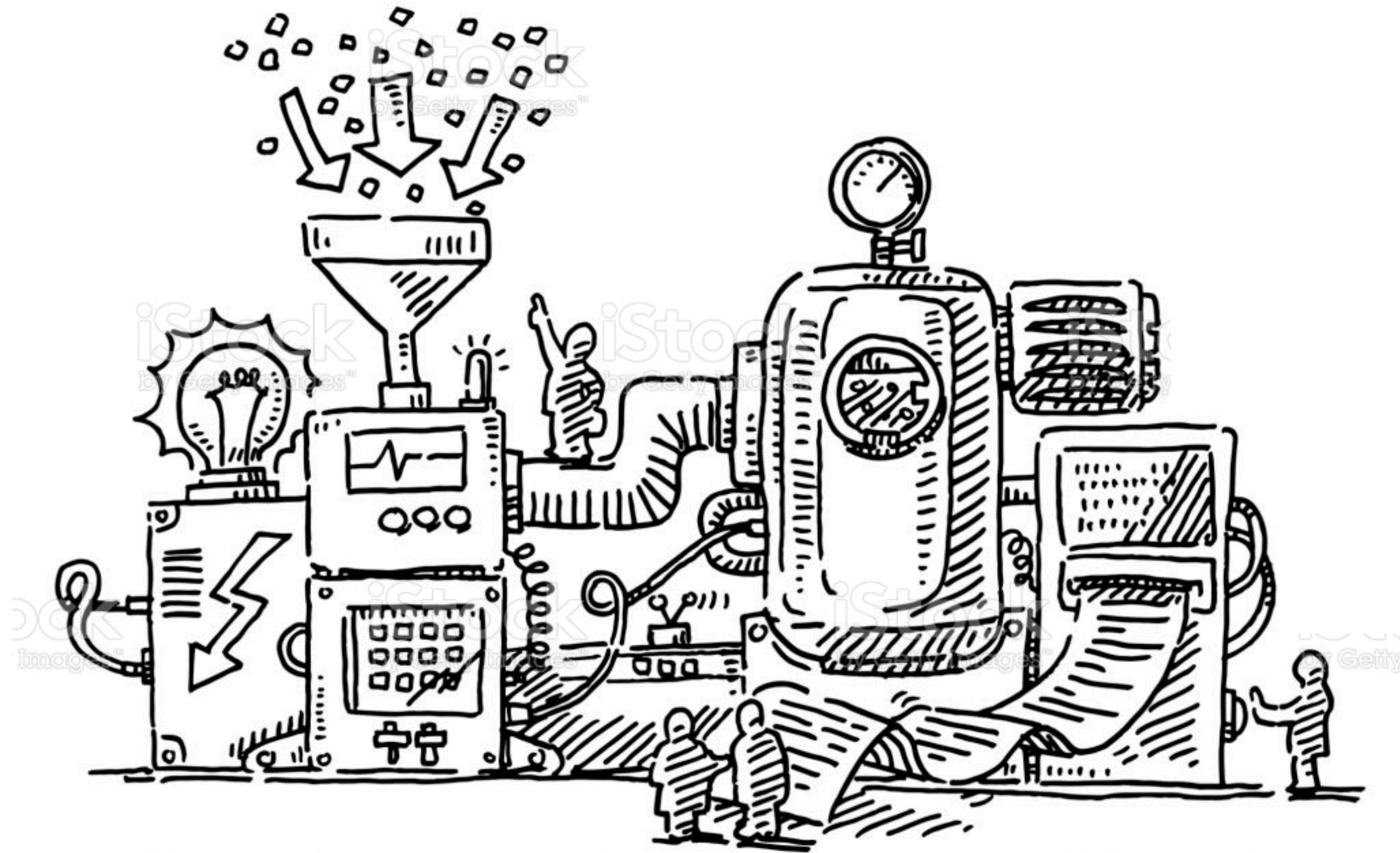


# BigPanda SNMP MIB Converter



**A customizable utility for creating BigPanda Event Configs**

# What it is...

- Utility for “Helping” generate BigPanda event configs.
- Its not a one-shot conversion tool and will likely require tuning/adjustments for each MIB
- Its primarily designed for converting large mibs that are too difficult to do by hand.
- Understanding of Javascript will be extremely helpful.
- Several examples are provided and in most cases only small changes are needed.

# The challenge with converting MIBS

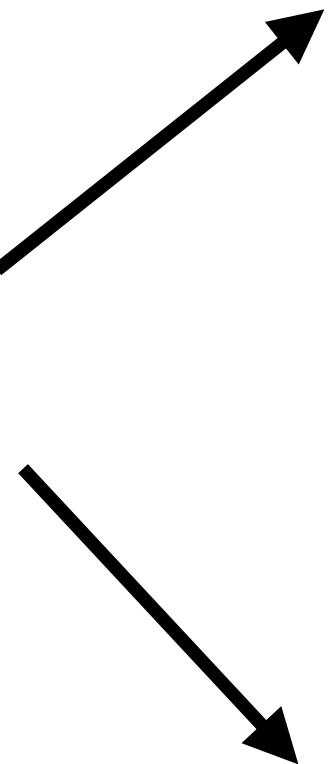
- SNMP standard gives vendors a lot of leeway in how they create MIBS:
  - inconsistent formatting
  - what's include in a trap definition
  - how traps are named
  - how status is relayed
  - what varbinds mean
  - etc...
- Converting very large MIBS with hundreds or thousands of traps

# Primary Tasks

- Identifying traps and setting primary & secondary properties
- Identifying trap status
- Making the description and other alert details meaningful
- Converting numerical values into something useful for humans

# Embedded Identity & Status

Trap Names



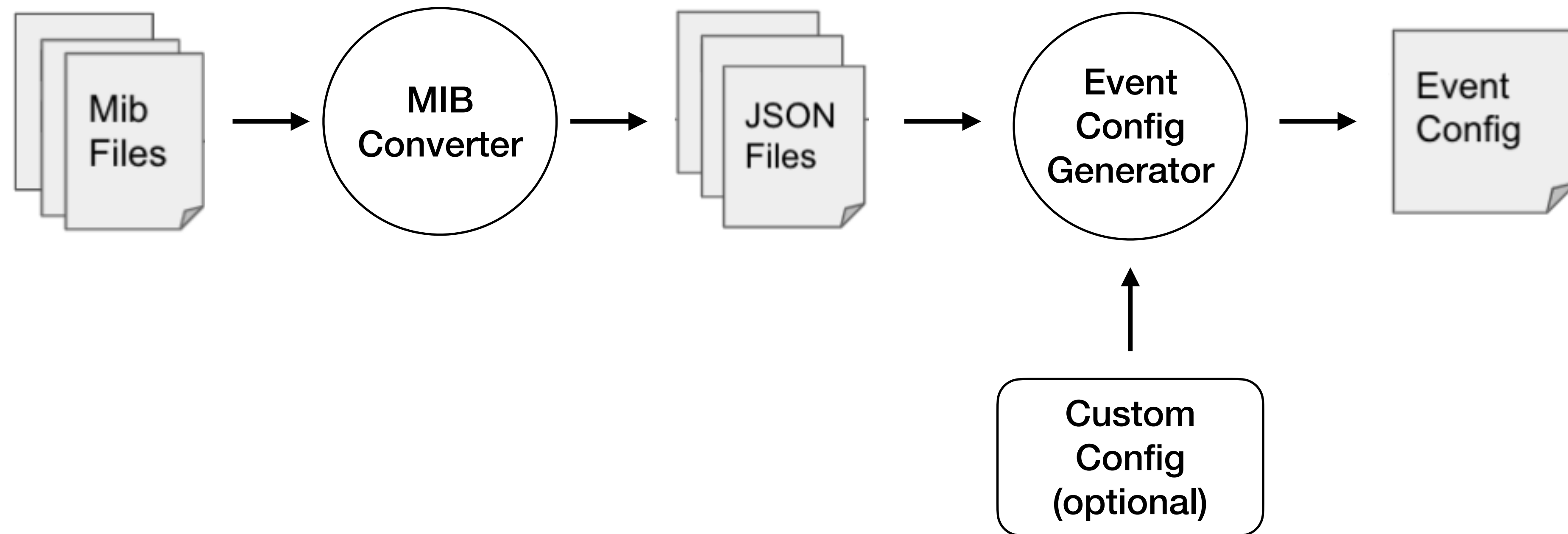
```
bigipNodeDown    NOTIFICATION-TYPE
OBJECTS {
    bigipNotifyObjMsg,
    bigipNotifyObjNode
}
STATUS          current
DESCRIPTION
    "A node is detected DOWN."
::= { bigipNotifications 12 }

bigipNodeUp      NOTIFICATION-TYPE
OBJECTS {
    bigipNotifyObjMsg,
    bigipNotifyObjNode
}
STATUS          current
DESCRIPTION
    "A node is detected UP."
::= { bigipNotifications 13 }
```

Example Substrings

```
"Fail":"critical",
"Bad":"critical",
"Info":"warning",
"Warning":"warning",
"On":"ok",
"Off":"warning",
"Start":"ok",
"Stop":"warning",
"Enabled":"ok",
"Disabled":"warning",
"Low":"warning",
"Warn":"warning",
"Change":"warning",
"ChangeReason":"warning",
"NotAvail":"critical",
"Avail":"ok",
"Online":"ok",
"WarnClear":"ok",
"AlarmClear":"ok",
"Clear":"ok",
```

# How it works...



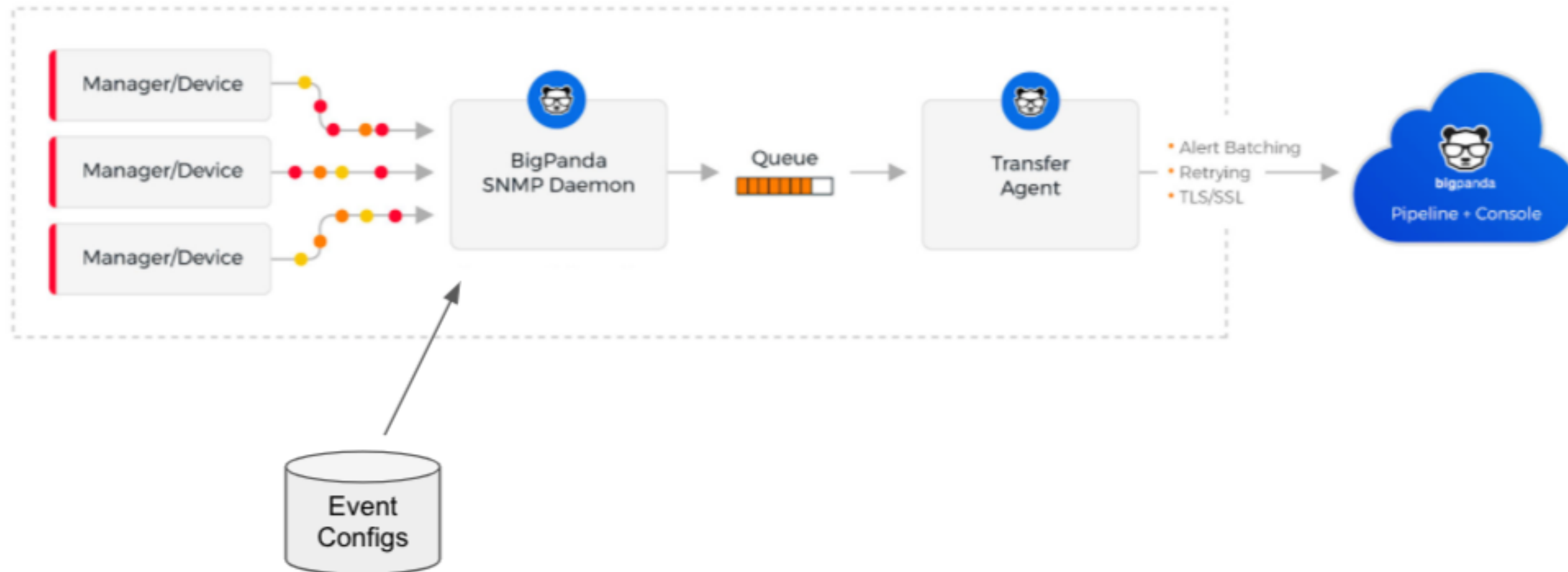
# Example Conversion Process

- 1) Create working directory and copy mib files to it
- 2) Run `convert.sh` to create a JSON version of the MIB
  - only needs to be done once
- 3) Run `generator.sh` to create the event config
  - May need to run several times to fine tune the generated content
- 4) Update the generated event config by hand if needed



# Installing...

- Copy generated event configs to /etc/bigpanda/snmpd/event\_configs
- Update /etc/bigpanda/snmpd/snmp-daemon.json to point to new files
- Restart the SNMP Agent





# Generator Configs

- Configuration files (Javascript Classes) are used to manipulate the generated output and can be customized as needed
- Elements that can be customized include:
  - Primary / Secondary Keys
  - Configuring Status Mapping / Conversion Logic
  - Copying and/or renaming var binds
  - Adding new tags

# Default Generator Class

```
module.exports = class generator {  
  ...  
  statusMapTags(obj) {  
    let o = {};  
    return o;  
  }  
  setTags(obj) {  
    let o = {  
      "status": "critical",  
      "description": obj.description ? obj.description : ""  
    };  
    return o;  
  }  
  copyTags(obj) {  
    let o = {};  
    return o;  
  }  
  renameTags(obj) {  
    let o = {};  
    return o;  
  }  
  primaryKeyValue(obj) {  
    return "snmp_source_ip";  
  }  
  secondaryKeyValue(obj) {  
    return "snmp_trap_name";  
  }  
}
```

# Example Custom Generator

```
module.exports = class generator {  
  ...  
  statusValue(obj) {  
    return "critical";  
  }  
  statusMapTags(obj) {  
    return {  
      "1": "ok",  
      "2": "warning",  
      "3": "warning",  
      "4": "critical",  
      "5": "critical"  
    };  
  }  
  setTags(obj) {  
    let o = {  
      "description": obj.description ? obj.description : "test-description"  
    };  
    return o;  
  }  
  renameTags(obj) {  
    let o = {  
      "nnmiIncidentSourceNodeHostname": "host",  
      "nnmiIncidentSeverity": "status"  
    };  
    return o;  
  }  
  primaryKeyValue(obj) {  
    return "host";  
  }  
}
```

# Example Custom Generator

```
const statusMap = require("../f5.json");
```

```
module.exports = class generator {
```

```
  setTags(obj) {
```

```
    let check = statusMap[obj.name] ? statusMap[obj.name].name : "Unknown";
```

```
    let status = statusMap[obj.name] ? statusMap[obj.name].status : "critical";
```

```
    let o = {
```

```
      "check": check,
```

```
      "status": status,
```

```
      "description": obj.description ? obj.description : "test-description"
```

```
    };
```

```
    return o;
```

```
  }
```

```
  renameTags(obj) {
```

```
    let o = {
```

```
      "snmp_source_hostname" : "host"
```

```
    };
```

```
    return o;
```

```
  }
```

```
  primaryKeyValue(obj) {
```

```
    return "host";
```

```
  }
```

```
  secondaryKeyValue(obj) {
```

```
    return "description";
```

```
  }
```

```
}
```

```
{  
  "bigipAgentStart": {  
    "name": "bigipAgent",  
    "status": "ok"  
  },  
  "bigipAgentShutdown": {  
    "name": "bigipAgentShutdown",  
    "status": "critical"  
  },  
  "bigipAgentRestart": {  
    "name": "bigipAgentRestart",  
    "status": "critical"  
  },  
  "bigipCpuTempHigh": {  
    "name": "bigipCpuTemp",  
    "status": "critical"  
  },  
}
```

# Example Custom Generator

```
module.exports = class generator {
  ...
  statusValue(obj) {
    if (obj.name.indexOf("Warn") !== -1) {
      return "warning";
    } else if (obj.name.indexOf("Ok") !== -1) {
      return "ok";
    }
    return "critical";
  }
  setTags(obj) {
    let o = {
      "status": this.statusValue(obj),
      "description": obj.description ? obj.description.replace(/\r\n/g, "").replace(/\s+/g, " ") : ""
    };
    return o;
  }
  primaryKeyValue(obj) {
    var name = "snmp_source_ip";
    if (obj.objects) {
      obj.objects.forEach(o => {
        if (o.object === "sysName")
          name = "sysName";
      });
    }
    return name;
  }
  secondaryKeyValue(obj) {
    return "snmp_trap_name";
  }
}
```



# Hints & Things to Watchout for

- Finding MIBS - If you don't have them already you'll need to find the source mibs. Best from the vendor, but you can also search online repos like [mibdepot.com](http://mibdepot.com)
- Inspect the MIBS for any dependent MIBs and get them too
- Dealing with multiple event configs for the same vendor.
  - Make sure you don't have duplicate trap names. The SNMP agent requires each name to be unique and will fail to start if it finds duplicates
  - It helps to combine the event configs and do some grep'ing and command line magic to find duplicates. I often grep for all the trap names and load them into excel where I can find duplicates.
- There have been some reported errors in how the trap OID is generated. Some time it contains (or is missing) and extra ".0" at the end.
- See our docs for information on SNMP Agent and event config formatting here: <https://docs.bigpanda.io/docs/snmp-simple-network-management-protocol#how-it-works>