Delivering improved service with Nastel AutoPilot
Summary
Enterprise Stakeholders

- **Middleware Team**
  - DEV, TEST & PROD
  - Management of messaging backbone

- **Application Support**
  - Faster time to repair (MTTR)
  - Identify root cause of MQ issues

- **Application Developers**
  - Self-Service tooling for Dev, QA & UAT testing
  - Improved quality of new releases

- **Enterprise Architects**
  - Improve processes
  - Reduce costs
  - Prevent performance problems

- **Application Owners**
Middleware Management - Challenges

- No single point of control for multiple middleware, multiple platforms
- Middleware teams unable to delegate authority to Dev/Ops
- No personalized views of infrastructure
- Application Development teams cannot provision needed middleware objects
- Application Development teams cannot easily test message flows
- Middleware teams challenged by software upgrades & migrations
Middleware Management - Nastel’s Solution

- Automated discovery of middleware estate
- Simplified configuration management
- Easily manage and rollback changes
- Full audit trail of changes (who, what, where)
- Full message management & search
- User group and role management
- Secure, granular delegation of specific authorities to Dev/Ops
Proactive Monitoring - Challenges

- Unmonitored application assets
- Incomplete views of application estate
- Monitoring via complex scripting
- Inability to see troublesome trends
- Multiple teams using diverse tools
- Difficulty to isolate root cause
- False alerts
- “War-room” syndrome
- Operational and reputational risk
Proactive Monitoring - Nastel’s Solution

**NASTEL**
- Easily managed, policy-based dashboards
- Elimination of false alerts
- Trend & pattern detection
- Notifications & proactive alerts

**PROACTIVE MONITORING**
- Improved MTBF
- Automated actions
- Improved application availability & reliability
- Reduced MTTR
- Reduction of operational risks
Inconsistent transaction performance
“Missing” orders | invoices | trades ...
No historical data or trend analysis
Inability to detect anomalies
No analytics in business context
Perishable time-series data
Regulatory compliance obligations
Service Level Agreements
Operational risks
Audit requirements
NASTEL

- Full visibility of business flows: resources, applications, events, timings, topologies
- Isolate root causes, facilitate triage
- Set objectives and manage compliance (SLA’s, regulations)
- View workflow performance in business context
- 360° Situational Awareness
Nastel Solution Summary

- **Nastel Xray**
  Predictive Analysis of your business applications using machine learning, ergonomic communications, advanced visualization and automation.

- **Nastel AutoPilot**
  Proactively & Holistically Monitor Infrastructure, Platform, Middleware and Applications.

- **Nastel Navigator**
  Manage Middleware, securely, efficiently and effectively.
“... Measure and improve customer experience by using next generation analytics and machine learning technology ...”
Nastel Technologies: What we do

IT Operational Analytics & Intelligence
- Detect anomalies, reduce MTTR
- Improve productivity, customer experience
- Powered by machine learning, latest technology

Application Performance Management
- Visibility into apps, transactions, business flows
- 360 degree Situational Awareness
- Performance, compliance (SLA’s, regulations)

Middleware Management & Monitoring
- Automated discovery & configuration management
- Message management & full audit trail
- Highly granular delegation of authority to Dev/Ops
<table>
<thead>
<tr>
<th>FINANCIAL SERVICES</th>
<th>HEALTHCARE</th>
<th>TELCO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Citi</td>
<td>McKesson</td>
<td>SKY</td>
</tr>
<tr>
<td>TD Bank</td>
<td>BlueCross BlueShield</td>
<td>BARCLAYS</td>
</tr>
<tr>
<td>BNY Mellon</td>
<td>UnitedHealth Group</td>
<td>CME Group</td>
</tr>
<tr>
<td>FannieMae</td>
<td>BNP PARIBAS</td>
<td>DTCC</td>
</tr>
<tr>
<td>Northwestern Mutual</td>
<td>Banamex</td>
<td>First Citizens Bank</td>
</tr>
<tr>
<td>GMF</td>
<td>BCP</td>
<td></td>
</tr>
<tr>
<td>Credit Suisse</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ENERGY</th>
<th>TRAVEL</th>
<th>WHOLESALE DISTRIBUTION</th>
<th>CHEMICALS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ameren</td>
<td>Lufthansa</td>
<td>Winsupply</td>
<td>Dupont</td>
</tr>
<tr>
<td>New York Power Authority</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Debenhams</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Why us?</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td><strong>Technology</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Scale</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Customer Focus</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
Use Cases
Common Business Challenges of Our Customers

**Financial Services** - Avoiding stiff penalties for failing to reconcile with Federal Reserve funds transfers deadlines

**Financial Services** - Reducing risk by tracking actual or potential breaches in trade compliance (e.g., Dodd-Frank trade reporting)

**Healthcare** - Tracking electronic health records and claims processing in healthcare insurance with strict adherence to HIPAA rules

**Retail** - Capturing of user experience data in real time from POS for improved business planning

**Telco** - Slashing support costs for on-line order management of services
Case Study: Large Bank

Problem: SLA & compliance violations with resultant penalties

- No visibility of messaging transactions
- No ability to locate “lost” transactions or identify “latent” transactions
- No ability to categorize transactions by business lines & asset groups
- Financial and reputational impact visible to senior management

Solution: AutoPilot transaction tracking, monitoring & analytics

- Across entire application stack: Mainframe, Unix, Linux, Java, IBM MQ
- End-to-end transaction monitoring and visibility
- “Business milestone” views understandable to business users
- Rapid identification of problem transactions and missed SLA’s
- Ability to prioritize messaging transactions by multiple criteria
Case Study: Retail Organization

Problem: Order fulfillment application problems

- Reputational damage due to delays in order fulfillment
- Lost orders & customer attrition
- Event monitoring system overwhelmed; ticket backlog at service desk
- Majority of problems required Tier 3 support ($$$)

Solution: AutoPilot CEP automation

- Replaced in-house “eyes-on-screen” event monitoring system
- Situational awareness, automated detection of anomalies
- Dramatically reduced order failures, improved customer experience
- Increased bottom-line revenue, reduced support costs
- Solution deployed & providing proven value in less than 2 months
Case Study: Telecom Organization

Problem: Web application issues and dissatisfied customers

- Poor order-entry web application causing customer dissatisfaction
- 80% of orders had problems or were lost
- Metrics for each system segment not shared between engineers
- Expensive Tier 3 & development staff spending time on troubleshooting
- No resources available for development of new services

Solution: AutoPilot predictive and prescriptive analytics

- Replaced in-house, reactive monitoring system with proactive analytics
- Dramatically reduced need for expensive Tier 3 and development staff
- Predictive analytics help to rapidly identify data patterns and root cause

Reduction in service issues: 30%
Reduction in Tier 3 support costs: 70%
Reduction in MTTR: 45%
Problems we have seen

- Messages routed incorrectly and no one knew where they were
- Auditors reviewed MQ Admin rights and found security gaps
- Company committed to delivery SLA and had to way to measure
- Cluster change caused message to loop between 2 queue managers
- Application browsed all messages on a queue, waited 1 second and repeated. At one point, there were 3000 messages on the queue
Natural query language

- Ability to ask any question about application performance, logs, transaction and metrics using JKQL English like query language.

JKQL query language provides a flexible way to organize, query and ask questions about any data from any source.

Real time (CEP/EPL) expressions that let App Support see what’s happening in real-time.
Revenue potentially lost due to performance problems

Break down of revenue loss by time and location

Dill down into all events responsible for revenue loss

Revenue loss and reasons why
Drilldown to graph of nodes and edges representing topology at time of anomaly

- Topology graph automatically created for all relationships in time window
- Related nodes are automatically stitched together
- Lines between nodes (called “edges”), represent relationships between nodes
Analyzing MQ Logs

Breakdown by QMGR

- Number of events where eventname starts with 'AMQ2' group by QMGR
  - Pie chart
  - V910Test: 100.00%

Breakdown by Message ID

- Number of events where eventname starts with 'AMQ2' group by eventname order by number of desc
  - Bar chart
  - AMQ01001: 80
  - AMQ07545: 70
  - AMQ05261: 60
  - AMQ0954W: 50
  - AMQ0525: 40
  - AMQ0954I: 30
  - AMQ0954E: 20
  - AMQ0954I: 10

MQ Error by Day

- Number of events where eventname starts with 'AMQ2' group by starttime bucketed by day
  - Bar chart
  - August 02: 120
  - August 04: 100
  - August 06: 80
  - August 08: 60
  - August 10: 40
  - August 12: 20
  - August 14: 0
  - August 16: 0
  - August 18: 0
  - August 20: 0

Event Search

- Number of events where eventname starts with 'AMQ2' group by severity
  - Scorecard
  - Severity: INFO, WARNING, ERROR
  - Events Count:
    - INFO: 410
    - WARNING: 25
    - ERROR: 58

©2018 NASTEL TECHNOLOGIES, INC.
### Diagnosing MQ Messages

The image shows a dashboard interface for diagnosing MQ messages. It includes a table with columns for **Queue Name**, **Current Depth**, **Maximum Depth**, **Get Messages**, **Put Messages**, **Open Input Counter**, **Open Output Counter**, and **Last Updated**. The table is titled **Local Queues** and includes details for various queues such as `HTTP_OUT`, `MyTrace.Out`, `Q2`, and `GOFN`. The interface also has a section for **Console** with options like `Q2`, `DLH`, `MDE`, `Data Size`, `MD-Type`, `MD-Format`, `MD::Message ID`, `MD::Correl. ID`, and `MD::Put Date`.

### Table Overview

<table>
<thead>
<tr>
<th>Queue Name</th>
<th>Current Depth</th>
<th>Maximum Depth</th>
<th>Get Messages</th>
<th>Put Messages</th>
<th>Open Input Counter</th>
<th>Open Output Counter</th>
<th>Last Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP_OUT</td>
<td>61</td>
<td>5000</td>
<td>Allowed</td>
<td>Allowed</td>
<td>0</td>
<td>0</td>
<td>00:29:08 hours</td>
</tr>
<tr>
<td>MyTrace.Out</td>
<td>24</td>
<td>5000</td>
<td>Allowed</td>
<td>Allowed</td>
<td>0</td>
<td>0</td>
<td>00:29:08 hours</td>
</tr>
<tr>
<td>Q2</td>
<td>19</td>
<td>5000</td>
<td>Allowed</td>
<td>Allowed</td>
<td>0</td>
<td>0</td>
<td>00:29:08 hours</td>
</tr>
<tr>
<td>HTTP_LISTENER_OUT</td>
<td>2</td>
<td>5000</td>
<td>Allowed</td>
<td>Allowed</td>
<td>0</td>
<td>0</td>
<td>00:29:08 hours</td>
</tr>
<tr>
<td>GOFN</td>
<td>1</td>
<td>5000</td>
<td>Allowed</td>
<td>Allowed</td>
<td>0</td>
<td>0</td>
<td>00:29:08 hours</td>
</tr>
<tr>
<td>VTest</td>
<td>0</td>
<td>5000</td>
<td>Inhibited</td>
<td>Allowed</td>
<td>0</td>
<td>0</td>
<td>00:29:08 hours</td>
</tr>
<tr>
<td>SampleReplyQueue</td>
<td>0</td>
<td>5000</td>
<td>Allowed</td>
<td>Allowed</td>
<td>2</td>
<td>0</td>
<td>00:29:08 hours</td>
</tr>
<tr>
<td>SampleOutputQueue</td>
<td>0</td>
<td>5000</td>
<td>Allowed</td>
<td>Allowed</td>
<td>0</td>
<td>0</td>
<td>00:29:08 hours</td>
</tr>
<tr>
<td>Q1</td>
<td>0</td>
<td>5000</td>
<td>Allowed</td>
<td>Allowed</td>
<td>0</td>
<td>0</td>
<td>00:29:08 hours</td>
</tr>
<tr>
<td>REQUEST.QUEUE</td>
<td>0</td>
<td>5000</td>
<td>Allowed</td>
<td>Allowed</td>
<td>0</td>
<td>0</td>
<td>00:29:08 hours</td>
</tr>
<tr>
<td>REPLY.QUEUE</td>
<td>0</td>
<td>5000</td>
<td>Allowed</td>
<td>Allowed</td>
<td>0</td>
<td>0</td>
<td>00:29:08 hours</td>
</tr>
</tbody>
</table>

**Total: 50 Visible, 60 Selected 8**

---

**Console Options:**
- `Q2`
- `DLH`
- `MDE`
- `Data Size`
- `MD-Type`
- `MD-Format`
- `MD::Message ID`
- `MD::Correl. ID`
- `MD::Put Date`

**Recent Actions:**
- Delete messages
- Copy messages
- More messages
Track & Trace Digital Transactions

1. Tracking Events
   - AutoPilot® Insight
   - ANALYTICS ENGINE
   - CEP

2. Correlation, Analytics
   - Transaction Views
   - Activities, Events
   - Metrics, Analytics

3. Actionable dashboards

DATA CAPTURE
- Java
- SOA
- Service Bus
- Messaging

GLOBAL TRANSACTION
Example Transaction: Financial Services

jKQL> Get relatives from CALYPSO for latest 2 minutes show as topology

SWIFT.OFAC.STATMENTS.CONFIRMATIONS

Avg 882ms
Count: 11
BANKS_TRADES.OUT

CALYPSO

Avg 2007ms
Count: 73
JMS://queue/confirmation

Avg 19082ms
Count: 14
TRAM

Avg 1ms
Count: 1
Avg 2377ms
Count: 61
OFAC/BSA Screening

Avg 699ms
Count: 2
BANKS_MT300.OUT

SWIFT.SCR
Transaction Trace Details

Get relatives from CALYPSO for latest 2 minutes show as topology

[Diagram showing network trace details with nodes labeled SWIFTCtrlparty@11.0.0.120, SWIFTCtrlparty@11.0.0.23, CREDITFUNDS, and connections to CALYPSO and MQBRIDGE.]

<table>
<thead>
<tr>
<th>ActivityName</th>
<th>StartTime</th>
<th>EndTime</th>
<th>ElapsedTime</th>
<th>Severity</th>
<th>EventCount</th>
<th>Snapshot</th>
<th>SourceFQN</th>
</tr>
</thead>
<tbody>
<tr>
<td>CALYPSO-CIBOS</td>
<td>2016-09-17 11:14:57.873000</td>
<td>2016-09-17 11:14:58.906568</td>
<td>1.033s</td>
<td>INFO</td>
<td>1</td>
<td>0</td>
<td>APPL=CALYPSO-CIBOS#SERVER=CALYPSO</td>
</tr>
<tr>
<td>CALYPSO-CIBOS</td>
<td>2016-09-17 11:15:01.415631</td>
<td>2016-09-17 11:15:01.415798</td>
<td>776ms</td>
<td>INFO</td>
<td>2</td>
<td>0</td>
<td>APPL=CALYPSO-CIBOS#SERVER=CALYPSO</td>
</tr>
<tr>
<td>CALYPSO-CIBOS</td>
<td>2016-09-17 11:14:58.849000</td>
<td>2016-09-17 11:15:03.048384</td>
<td>4.199s</td>
<td>INFO</td>
<td>1</td>
<td>0</td>
<td>APPL=CALYPSO-CIBOS#SERVER=CALYPSO</td>
</tr>
</tbody>
</table>

©2017 NASTEL TECHNOLOGIES, INC.
Transaction Analytics, Notifications and Alerts

Get relatives from CALYPSO for latest 2 minutes show as topology

<table>
<thead>
<tr>
<th>EventName</th>
<th>StartTime</th>
<th>ElapsedTime</th>
<th>EventType</th>
<th>Message</th>
<th>Severity</th>
<th>Exception</th>
<th>ResourceName</th>
</tr>
</thead>
<tbody>
<tr>
<td>/book-trade</td>
<td>2016-09-23 17:42:57.382000</td>
<td>0ms</td>
<td>OTHER</td>
<td></td>
<td>INFO</td>
<td></td>
<td>book-trade</td>
</tr>
<tr>
<td>HTTP/POST/book-trade</td>
<td>2016-09-23 17:42:57.382460</td>
<td>0ms</td>
<td>OTHER</td>
<td>(:15A:20:LD0006...</td>
<td>INFO</td>
<td></td>
<td>book-trade</td>
</tr>
<tr>
<td>QueueSender.send</td>
<td>2016-09-23 17:42:57.382938</td>
<td>13ms</td>
<td>SEND</td>
<td>(:15A:20:LD0006...</td>
<td>INFO</td>
<td></td>
<td>JMS/queue/tradein</td>
</tr>
<tr>
<td>MQPUT</td>
<td>2016-09-23 17:42:57.396887</td>
<td>2.162s</td>
<td>SEND</td>
<td>(:15A:20:LD0006...</td>
<td>FAILURE</td>
<td>2010:MQRC_DATA...</td>
<td>BANKS TRADES IN QUEU</td>
</tr>
</tbody>
</table>

©2017 NASTEL TECHNOLOGIES, INC.
### Transaction Activity Trace, Event and Message Details

<table>
<thead>
<tr>
<th>StartTime</th>
<th>EndTime</th>
<th>ElapsedTime</th>
<th>EventCount</th>
<th>AppName</th>
<th>ResourceName</th>
</tr>
</thead>
<tbody>
<tr>
<td>2016-09-17 11:14:57.873000</td>
<td>2016-09-17 11:14:59.150676</td>
<td>1.277s</td>
<td>2</td>
<td>CALYPSO-CIBOS.TRAM</td>
<td>BANKS.MT300 DUP5 TO TRAM QUEUE.BAN</td>
</tr>
<tr>
<td>2016-09-17 11:14:57.932772</td>
<td>2016-09-17 11:14:59.154368</td>
<td>1.231s</td>
<td>23</td>
<td>OFAC/BSA Screening, CALYPSO-CIBOS, M</td>
<td>ATP001.QMGR BANKS.MT300 DUP5 TO TRAM QUEUE.BAN</td>
</tr>
<tr>
<td>2016-09-17 11:15:00.765868</td>
<td>2016-09-17 11:15:00.765868</td>
<td>2.582s</td>
<td>3</td>
<td>CALYPSO-CIBOS.TRAM</td>
<td>BANKS.MT300 DUP5 TO TRAM QUEUE.BAN</td>
</tr>
<tr>
<td>2016-09-17 11:15:01.116293</td>
<td>2016-09-17 11:15:01.116293</td>
<td>2.978s</td>
<td>23</td>
<td>OFAC/BSA Screening, CALYPSO-CIBOS, M</td>
<td>ATP001.QMGR BANKS.MT300 DUP5 TO TRAM QUEUE.BAN</td>
</tr>
</tbody>
</table>

### JQQL Query

Get Events where ActivityID in ('3d978bd8-7cb7-11e6-b5de-00259035f22d') order by starttime asc

<table>
<thead>
<tr>
<th>EventName</th>
<th>StartTime</th>
<th>ElapsedTime</th>
<th>EventName</th>
<th>StartTime</th>
<th>ElapsedTime</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>/book-trade</td>
<td>2016-09-17 11:14:59.621000 +02:00</td>
<td>0ms</td>
<td>HTTP/POST/book-trade</td>
<td>2016-09-17 11:14:59.621446 +02:00</td>
<td>0ms</td>
<td>OTHER</td>
</tr>
<tr>
<td>QueueSender.send</td>
<td>2016-09-17 11:14:59.621924 +02:00</td>
<td>13ms</td>
<td>MQPUT</td>
<td>2016-09-17 11:14:59.635724 +02:00</td>
<td>2.507s</td>
<td>WARNING</td>
</tr>
<tr>
<td>MQGET</td>
<td>2016-09-17 11:15:02.117492 +02:00</td>
<td>26ms</td>
<td>MQPUT</td>
<td>2016-09-17 11:15:02.732387 +02:00</td>
<td>24ms</td>
<td>OTHER</td>
</tr>
<tr>
<td>QueueReceive.receive</td>
<td>2016-09-17 11:15:02.760401 +02:00</td>
<td>13ms</td>
<td>HTTP/Response/match-conf</td>
<td>2016-09-17 11:15:02.774201 +02:00</td>
<td>0ms</td>
<td>OTHER</td>
</tr>
</tbody>
</table>

Message Example:

```
```
Transaction Milestones: Financial Services Example

Get relatives from CALYPSO for latest 2 minutes show as topology

TRAM DUP CONF

CTR PARTY CONF

CALYPSO

OFAC

SAA

SCR CONF

TRAM MATCH

CONF MATCH

MT202

MT910/950

SAA PAYMNT

SCR PAYMNT

PAYMNT CMPLT
Transaction Milestones: Retail Example

Get relatives from CALYPSO for latest 2 minutes show as topology.

Order Placed > Verify Order > Verify Credit > Order Router > Process Payment

Process Payment

Activities (21)

Objectives

SLA
-ElapsedTime <= 2 seconds
  - Met (20)
  - Not met (1)

Successful
- CompCode = "SUCCESS"
  - Met (6)
  - Not met (15)
Automatic Detection of Anomalies

Out of band (anomaly)
Missed SLA Transactions

jKQL> Get events for today where ResourceName contains 'Missed SLA' show as linechart

Missed SLA Events

ElapasedTime: 00m 00.033593s

1049 Missed SLA Events

Severity | Event Count
---|---
INFO | 565
WARNING | 344
ERROR | 120

<table>
<thead>
<tr>
<th>EventName</th>
<th>StartTime</th>
<th>ElapsedTime</th>
<th>EventType</th>
<th>Message</th>
<th>Severity</th>
<th>SourceFQN</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReadOrder</td>
<td>2016-05-24 21:30:35.548116 +02:00</td>
<td>25ms</td>
<td>RECEIVE</td>
<td>Process Payment ProductId=8380203...</td>
<td>WARNING</td>
<td>APPL=Proces</td>
</tr>
<tr>
<td>ReadOrder</td>
<td>2016-05-24 21:30:48.542458 +02:00</td>
<td>27ms</td>
<td>RECEIVE</td>
<td>Process Payment ProductId=8380203...</td>
<td>WARNING</td>
<td>APPL=Proces</td>
</tr>
<tr>
<td>ReadOrder</td>
<td>2016-05-24 21:30:35.548116 +02:00</td>
<td>25ms</td>
<td>RECEIVE</td>
<td>Process Payment ProductId=8380203...</td>
<td>ERROR</td>
<td>APPL=Proces</td>
</tr>
</tbody>
</table>

©2017 NASTEL TECHNOLOGIES, INC.
Using IBM Activity Trace to better understand your application logic

IBM introduced Activity Tracing with the MQ appliance and is now available on distributed releases of MQ. Activity traces can be used for a variety of use cases. For example, they allow the MQ administrator to understand MQ and application behavior. They also allow application developers and support to determine if the expected MQ calls are being performed. Version 9 of MQ added subscription support which allow further control on who can request and display traces. This session will cover how to activate traces, tips to avoid some of the pitfalls you can encounter, and using IBM sample programs to analyze the data collected.
Tracking messages in your middleware environment with Nastel X-Ray

Being able to track messages as they move through MQ provides a wealth of information. While MQ statistics provide summary information, they often only tell half of the story. With tracking, you are able to see detailed MQ requests that explain the exact behavior of your applications. While using standard MQ tooling provides a good place to start, more complex flows require a more robust tracking tool. In this session, you will learn how Nastel X-RAY allows you to answer questions such as are applications following the coding standards, is the application call efficient, what caused messages to end up in the dead letter queue, and many others.
If you can’t clone yourself, Delegate!

You already have more work to do than you have time to complete and your organization wants you to do more. With Nastel AutoPilot OnDemand, you can delegate work to the responsible teams. It’s not simply a matter of giving everyone access to MQ. You need a secure solution that gives just enough rights to get the job done. You need to satisfy the auditors when they come knocking at your door. You need to manage MQ as well as the other middleware you have in place. You need a solution that is easy to use and scales to the needs of your organization. In this session, come see how Nastel AutoPilot OnDemand provides these capabilities and more.