independIT Integrative Technologies GmbH

New Features Release 2.11

2. Januar 2024

Ankündigung

Zope 2 Frontend Da das Zope 2 Frontend auf Python 2 basiert und diese Version von Python seit längerem nicht mehr supported wird, werden wir den Support für Zope 2 ab der nächste Release (2.12) iebenfalls einstellen. In der aktuellen Release 2.11 ist noch ein SDMS.zexp und einige Python-Skripte enthalten. Damit kann auch eine bestehende Zope 2 Installation mit dem 2.11-er Server weiterarbeiten. Eine Neuinstallation von Zope 2 ist allerdings nicht mehr möglich.

Neue Features

Neues Frontend Es wurde von Grund auf ein neues Frontend auf Basis von NodeJS entwickelt. Die browserbasierte Anwendung ist als Single Page Application konzipiert worden. Damit wurde ein wichtiger Nachteil des Zope-basierenden Frontends eliminiert. Zudem ist die Bedienung aufgrund fehlender Round-Trip Dialogschritten deutlich flüssiger und angenehmer. Auch das Aussehen der Anwendung wirkt deutlich moderner.

In dieser Release wird die neue GUI noch als Alpha-Software ausgeliefert. Auch wenn intensives Testen vielen der Problemchen aufgedeckt hat, kann nicht ausgeschlossen werden, dass es einige Funktionen gibt die nicht zu 100% funktionieren. Über Fehlermeldungen freuen wir uns sehr. Deshalb sollte zumindest als Fallback-Lösung weiterhin das Zope-basierende Frontend zur Verfügung stehen.

Approval System Auf Kundenwunsch wurde ein System für das Ermöglichen eines 4-Augen Prinzips für kritische Operator-Eingriffe implementiert. Auf Master-, Parent- oder Jobebene kann festgelegt werden, welche Operator-Eingriffe nach dem 4-Augen Prinzip durchgeführt werden, welche nur hinterher überprüft werden sollen, und welche unkritisch sind.

Wird das 4-Augen Prinzip gefordert, dann wird im Falle eines entsprechenden Operator-Eingriffs ein Approval-Request ins System eingestellt. Eine zweite Person mit dem Approval-Privileg kann die Aktion danach entweder bestätigen, oder ablehnen. Erst nach der Bestätigung durch die zweite Person, wird die Operation auch tatsächlich durchgeführt.

Wird nur eine anschliessende Überprüfung gefordert, wird die Operation sofort durchgeführt, aber es wird auch ein Approval Request ins System eingestellt. Anschliessend kann eine zweite Person die Operation gutheissen, oder ablehnen. Hierbei steht der dokumentative Charakter im Vordergrund.

Es kann zwischen den unterschiedlichen Operationen differenziert werden. Etwa ein CANCEL kann das 4-Augen Prinzip fordern, aber ein RERUN oder das setzen des Exit Status nicht.

Triggered By Row Limit Manche Jobs werden von sehr viele andere Jobs getriggert. Dies gilt zum Beispiel für Jobs die im Fehlerfall E-Mails oder andersartige Benachrichtigungen verschicken.
Es ist nun möglich im LIST TRIGGER Statement ein Limit zu spezifizieren um nur einen Teil der Trigger anzuzeigen. Dies erhöht die Performance maßgeblich.
Im Frontend ist generell ein Limit aktiv, TB_ROW_LIMIT, mit als Default 100.

ps Utility für Windows Um die Liste aller Prozessen mit ihren Startzeiten zu ermitteln wurde das standard Windows Programm WMIC benutzt. Es hat sich jedoch herausgestellt, dass es dabei Probleme bei der Umstellung von Sommerzeit nach Winterzeit oder umgekehrt gibt.
Eine Alternative zu WMIC wurde mit Hilfe der PowerShell geschaffen. Diese Alternative war jedoch sehr langsam und führte zu einem exzessiven CPU-Verbrauch.
Daher wurde eine dritte Lösung, "'winps", in Anlehnung an das *ps* Utility unter Unix, in C geschrieben. Diese Lösung hat weder Probleme mit der Umstellung von oder nach DST und ist genau genommen noch schneller als WMIC.

Besseres Frontend Support für das Object Monitoring Watch Types und Object Monitors können von der GUI aus definiert werden. Dazu wird das manage_watchtype privileg benötigt.

Detaillierte Operator Privilegien Zusätzlich zum einzelnen OPERATE Privileg, sind nun auch detaillierte Privilegien möglich. Das reine OPERATE Privileg entspricht dabei der Gesamtheit aller Einzelprivilegien, und ist damit semantisch gleichbedeutend mit dem OPERATE Privileg aus den vorherigen Releases. Das OPERATE Privileg kann aber weggenommen und dafür Privilegien für Einzeloperationen wieder vergeben werden. Damit kann z.B. verhindert werden, dass jeder Operator Jobs canceln können.

Modify Parameter Privileg protect set parameter by checking the MODIFY PARAMETER privilege Detailed operator privileges Web GUI implementation

Build Hash und Build Date Da bei Support Anfragen häufig nach dem Build Hash gefragt wird und diese etwas umständlich ermittelt werden musste, wird der Build Hash nun beim Show System ausgegeben. Und da der Hashwert für Menschen nicht besonders aussagekräftig ist wird ebenfalls das Build Datum mit ausgegeben.

Java Kompatibilität Neuere Java Versionen (11, 13, 17 und höher) werden jetzt auch unterstützt. Allerdings muss bei Benutzung von Java 17 und höher der Parameter MEMFLAGS aus der java.conf Datei geändert werden.

Die Einstellung

```
MEMFLAGS="-XX:+UseZGC -XX:-ZUncommit"
```

hat sich bis jetzt bewährt. Für Anregungen und Kommentare sind wir, wie immer, dankbar.

Sichtbarkeit von Intervallen Unabhängige Intervalle, das sind Intervalle die nicht zu irgendeiner Job Definition gehören, sind nun sichtbar und benutzbar für jeden. Selbstverständlich sind die andere Privilegien, DROP und EDIT, nicht für die Allgemeinheit freigegeben.

Unresolved Parameter Es ist nun konfigurierbar wie der Server mit problematischen Parametern umgehen soll. Fall ein Parameter nicht gefunden wird, wird entweder ein Fehler ausgegeben, oder es wird ein Leerstring als Wert zurückgegeben, oder der Parameter wird erst gar nicht ersetzt.
Der Konfigurationsparameter heisst *UnresolvedParameterHandling* hat hat folgende mögliche Werte:

- ERROR – Der Default, ein nicht-auflösbarer Parameter wird als Fehler betrachtet
- BLANK – Ein nicht-auflösbarer Parameter wird, ähnlich wie in einer Shell, mit einem Leerstring ersetzt
- ECHO – Der Parameter wird nicht ersetzt

jsctl Utility replace JSCTL.BAT with jsctl.exe

Show Interval Beim Show Interval Kommando können optional für eine Periode die generierte Blöcke ausgegeben werden. Das erleichtert die Entwicklung komplexer Intervalle.

Expand Rule Es wurde im Dump ein Expand Rule für das Enable Interval, aus der Parent-Child Beziehung zweier Job Definitions, zugefügt.

Ticketinterval konfigurierbar Das TicketInterval ist jetzt konfigurierbar. Dies kann die Übernahme des Betriebs durch einen Warm Standby Server beschleunigen, da dieser drei mal ein TicketInterval abwartet. Die minimale Zeit zwischen zwei Ticker-Updates beträgt drei Sekunden. Der Default Wert ist jetzt 20 Sekunden.

Broken Finished Handling Ein Serverparameter BrokenFinishedHandling mit den möglichen Werten RESTRICTED und CLASSICAL wurde zugefügt um das Verhalten des Servers im Falle eines BROKEN_FINISHED Status eines Jobs zu definieren. CLASSICAL ist das ursprüngliche Verhalten, was BROKEN_FINISHED als Broken State betrachtet hat. In dem Fall wird der Exit Status des Jobs aus den Broken State gesetzt, was wiederum weitere Verarbeitungen zur Folge haben kann. Im Falle RESTRICTED wird der Exit Status nicht (automatisch) gesetzt. Dies liefert mehr Kontrolle über das was passiert.

List Calendar LIST CALENDAR kann jetzt auch nach Owner filtern

scrolllog options Die Semantik der *-f* Option hat sich geändert. Log Output wird nicht nur nach den Logdateien sonder auch nach stdout geschrieben. Diese Option wird von *sdmsctl run* benutzt. Damit kann, zu Debug-Zwecken, der Server im Vordergrund gestartet werden.

Show Folder und Show Scope Parameteranzeige Das Show Folder und Show Scope Kommando zeigt nun alle sichtbare Parameter, insbesonder auch die Parameter die auf Parent-Ebene (oder höher) definiert sind.

Dump Language Level Das Dump Language Level beim Export aus der GUI oder mittels sdmsh kann nun spezifiziert werden. Damit können Dumps erzeugt werden die in früheren Releases eingelesen werden können. Es spricht für sich, dass neuere Features dabei nicht genutzt werden können. Es kann von daher auch nicht zu 100% garantiert werden, dass der Dump eines Ablaufs auch im älteren System einwandfrei funktioniert.

Parameter EXITCODE EXITCODE wurde als special Parameter for Jobs zugefügt. Der kann zum Beispiel für genauere Fehlermeldungen in Notifications hergenommen werden.

Neuer Parametertyp Der Paramnetertyp IMPORT_UNRESOLVED wurde zugefügt. Diese Parameters werden im Kontext des importierenden Jobs ausgewertet, nicht, wie bei IMPORT, im Kontext des definierenden Jobs.

List scheduled Kommando Es wurde ein neues Kommando *LIST SCHEDULED* implementiert. Damit kann für die angegebene Periode ermittelt werden, wann welche Batches und Jobs ausgeführt werden sollen. Wie im Calendar auch kann nach Jobname gefiltert werden.

Disabled Jobservers Ein Jobserver kann zu jeder Zeit disabled werden. Allerdings wird er sich weiterhin mit dem Server verbinden können, solange bis keine aktive Jobs mehr übrig sind. Danach ist der Zugang gesperrt.

Jobserver connected Anzeige Online Jobservers werden sofort rot angezeigt, wenn sie keine Serververbindung haben. Dies eliminiert die gelegentlich irreführende Anzeige, bei der die GUI den Jobserver noch eine Zeit lang grün angezeigt hat, in der Erwartung, dass der Jobserver sich sowieso gleich wieder verbindet.

Kompakte Liste mit Resource State Mappings Das Statement LIST RESOURCE STATE MAPPING FOR *profileName*; zeigt nur die Mappings, die mit dem spezifizierten Resource State Profile kompatibel sind.

Listen Interface Bislang wurde immer auf allen Interfaces gehört. Mit dem Parameter *Interface* bzw. *SSLInterface* kann ein spezifisches Interface angegeben werden.

sdmsh komfort Feature In *sdmsh* ist es jetzt in einem Multicommand erlaubt das letzte Statement nicht mit einem Semicolon abzuschliessen. sdmsh erkennt dies und fügt automatisch das fehlende Zeichen ein. Laut der offiziellen Syntax der Sprache ist der Semicolon jedoch weiterhin notwendig. Das Feature dient somit ausschliesslich der komfortablen Umgang mit sdmsh.

Erweiterung vom Connect Kommando Für die Protokolle JSON, PYTHON und PERL kann der Zusatz ZERO TERMINATED spezifiziert werden. Die Outputstruktur die als Antwort auf einen Kommando zurückgegeben wird, wird dann mit einem Null-Byte abgeschlossen. Diese Erweiterung erleichtert das Einlesen der Antwort maßgeblich.

Condensed Output Die Auflistung der Job Definition Hierarchie und Dependency Hierarchie kann gekürzt erfolgen. Dabei werden die Spalten, die im Allgemeinen nicht benötigt werden, aus der Ausgabe entfernt. Der Resultat ist eine bessere Performance aufgrund der verringerte zu verarbeitenden Datenmenge.

IP Adresse des Servers Wenn ein Warm-Standby Betrieb gefahren wird, d.h. es wird ein zweiter Server gestartet, der sofort übernimmt wenn der aktive Server ausfällt, ist es nicht immer offensichtlich, welcher Server gerade als aktiv gilt. In der Tabelle REPOSITORY_LOCK wird nun die IP-Adresse des aktiven Servers gespeichert. Auch in versehentlich herbeigeführte Fehlersituationen (etwa ein Fehler in der Datenbankkonfiguration) kann diese Information zu einer schnellen Fehlerbehebung beitragen.

Jobserver Konfiguration Es gibt drei weitere Parameter für die Jobserver Konfiguration.

In manchen Umgebungen funktioniert der Abgleich mit den Startzeiten der Prozessen nicht optimal. Es passiert dann gelegentlich, dass die ermittelte Startzeit zu sehr von der realen Startzeit abweicht. Die Folge ist, dass der Job in den Status BROKEN_FINISHED versetzt wird, obwohl er noch läuft. Durch das erhöhen der Toleranz beim Vergleichen der beiden Startzeiten kann dies verhindert werden. Der Parameter STARTIME_JITTER, per Default 5 Sekunden, kann dazu auf einen höheren Wert gesetzt werden. Wird er auf 0 gesetzt, wird die Startzeit gar nicht mehr berücksichtigt. Jeder Prozess mit derselben PID als der ursprüngliche Prozess wird als der zu überwachenden Prozess interpretiert. Dies ist meistens harmlos. Um eine Fehlinterpretation zu verursachen, muss der PID innerhalb sehr kurzer Zeit vom Betriebssystem erneut vergeben werden.

Durch unterschiedlichen Auffassungen verschiedener Betriebssystemen über die korrekte Terminierung einer Zeile (Unix sieht nur ein Linefeed vor, Windows benötigt auch noch ein Carriage Return), kann es sein, dass die falsche Terminierung im RUN_PROGRAM benutzt wird. Dies führt dann zu Fehlern bei der Interpretation der Argumente. Da der Jobserver entweder auf ein Unix-like oder auf ein Windows System (aber nie beides) läuft, kann der Jobserver eine Konvertierung, entweder nach Linefeed oder nach Carriage Return Linefeed vornehmen. Dazu wird der Parameter CONVERT_NEWLINE auf 0 gesetzt, findet keine Ersetzung statt. Der Wert 1 führt zu der Konvertierung nach Linefeed. Der Wert 2 wird für die Konverierung nach Carriage Return Linefeed benutzt.

Der dritte Parameter, HTTP_INTERFACE, dient dazu festzulegen auf welches Interface der Jobserver hört für den Logfile Display. Per Default wird auf allen Interfaces gehört. Mit Hilfe des genannten Parameters kann das auf ein spezifisches Interface reduziert werden.

Rename Parameters Wenn im Alter Job Kommando Parameter gesetzt werden, können diese umbenannt werden durch Angabe des Parameter IDs. Dadurch bleiben Referenzen erhalten (REFERENCE und CHILDREFERENCE Parameter).

Cancel with Kill Es ist möglich zum Cancel die Kill Option zu spezifizieren. Alle laufende Jobs mit Kill Program werden dann zuerst gekilled und anschliessend gecancelt. Zur Implementierung wurde ein Kill recursive benötigt. Diese Operation kann auch ohne Cancel benutzt werden.

UTF-8 Encoding für Log Output Beim rausgeben von Log Output wird der Text nach UTF-8 konvertiert. Dies ermöglicht die korrekte Anzeige von special Characters.

Audit Set Parameter Wenn ein Operator ein Paremeterwert eines Jobs setzt, wird die im Audit Trail festgehalten.

Behobene Fehler

Anführungszeichen bei Parameter und Trigger Namen Fehlende Anführungszeichen bei Trigger und Parameter Namen führten zu Problemen bei Namen mit Kleinschreibung und Sonderzeichen. Diese Namen werden nun korrekt gequotet.

Upgrade Skripte verbessert Die generated_upgrade Skripte enthalten jetzt keine drop Statements mehr für Views die es erst in neueren Releases geben wird. Es wird auch nicht versucht Diese Views anzulegen, wenn die Basistabellen noch gar nicht vorhanden sind.

Zope 5 Support Es wurden einige Inkompatibilitäten mit neueren Zope 5 Releases entfernt.

Resource Schedule als INTERNAL Wenn während des Resource Scheduling, ausgelöst durch eine Get Next Job Anfrage eines Jobservers, Auditeinträge gemacht werden, dann wird dies als User INTERNAL ausgeführt.

Parameter Resolution In manchen (fehlerhaften) Konstruktionen von aufeinander verweisenden Parameter kam es zu undeutlichen Fehlermeldungen. Diese wurden verbessert

Copy Job Definition Beim Kopieren von Job Definitions wurden die Kommentare nicht mitkopiert. Dieser Fehler wurde behoben.

Fehlender Status im Exit State Profile Falls der Timeout Status nicht im Exit State Profile eines Jobs enthalten ist, wird der Job auf ERROR gesetzt und keine Exception mehr geworfen.
Falls der Limit State nicht im Exit State Profile eines Jobs enthalten ist, wird der Job auf ERROR gesetzt. Dabei wird der Broken State des Exit State Profiles nicht gesetzt, da dies wiederum dazu führen kann, dass der Trigger, der sein Limit erreicht hat, wieder aktiviert wird.

Korrekte Behandlung fehlerhafter Syntax Unterminierte Parameterreferenzen am Ende eines Strings, etwa '${PARM', führten zu einem ArroayOutOfBounds Exception.

Leere SELECT Listen Eine leere SELECT Liste ist in manchen Datenbanksystemen gültig, fühte aber zu einem Absturz von Server. Dies wird nun richtig abgefangen.

Nice Profiles Beim Aktivieren oder Deaktivieren von Nice Profiles werden Masters in einem CANCELLING Status jetzt richtig behandelt.

Buffer für PATH Variable Es wurde nur 1KB Platz für den Inhalt der PATH Variable bereitgestellt. Dies wurde auf 8KB erhöht.

Dump Kommando Quoted null Values werden im Dump als Leerstring gerendert.

Commit am Ende eines Upgrades Am Ende der generated_upgrade Skripte wird nun ein Commit Statement geschrieben um zu verhindern, dass manche Datenbankengines die komplette Transaktion (das Skript also) zurückrollen, weil das Commit fehlt.

Race Condition Wenn ein BROKEN_FINISHED Situation detektiert wird, wird eine Sekunde gewartet um auszuschliessen, dass die Feststellung fälschlicherweise gemacht wurde.

Edit Resource Requests In einer Resource Requirement kann ein vorhandenes Resource State Mapping auch wieder entfernt werden.

Quote Identifiers In den SQL Statements werden jetzt alle Identifiers gequoted um Probleme mit neuen SQL Keywords zu vermeiden.

Create User Beim Anlegen eines Benutzers wird 1 ms gewartet. Damit wird erreicht, dass auch beim maschinellen Anlegen der generierte Salt für jeden Benutzer unterschiedlich ist.

Zombies unter Windows Verhindere die Enstehung von Zombies unter Windows; Referenzen auf Prozesse werden explizit gelöscht um Zombies zu verhindern

Verberge gelöschte Objekte Gelöschte Gruppen sowie gelöschte Benutzer werden nicht mehr angezeigt.