

ATLAS 実験データ収集システムに おけるイベントビルダ モニタリングシステムの開発

広島工大工 福田宏見 長坂康史
信州大理 長谷川庸司
長崎総科大情報 下島真

目次

- 背景
- 研究目的
- TDAQ システム
 - ・ Online Software システム
 - ・ Online Software フレームワーク
 - ・ IGUI
- Event Builder Panel
- シミュレーション
- まとめ

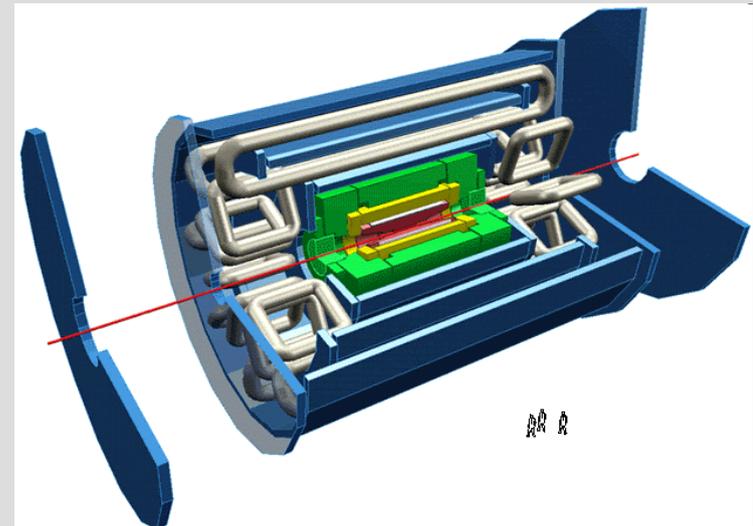
背景

■ ATLAS 実験

- CERN（欧州合同原子核素粒子研究所）

■ 測定器

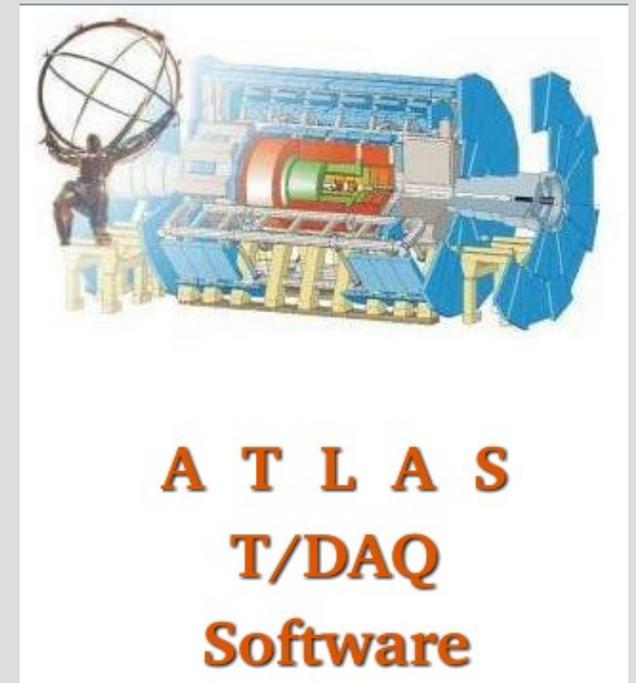
- 高さ 25 メートル
- 全長 44 メートル
- 重量 7000 トン



背景

■ ATLAS 実験のためのデータ収集システム

- ・ TDAQ システム : ATLAS Trigger and Data Acquisition System
- ・ 数千ノードで測定された測定結果を収集
- ・ 一つの意味のあるイベントデータに再構築



背景

■ TDAQ システム

- ・ 数千のノードで構成
- ・ 一つ一つ個別に管理するのは効率が悪い
- ・ イベント発生頻度：約 40MHz の頻度でイベントが発生
 - ・ フィルタリング後：約 3kHz
- ・ イベントサイズ：約 1MB

■ システム全体の状態を可視化

- ・ システムの状態監視、分析が容易

研究目的

■ 既存の TDAQ システム

- ・ 必要最小限の機能のみ
- ・ GUI アプリケーション
- ・ 個々のノードの状態を表示

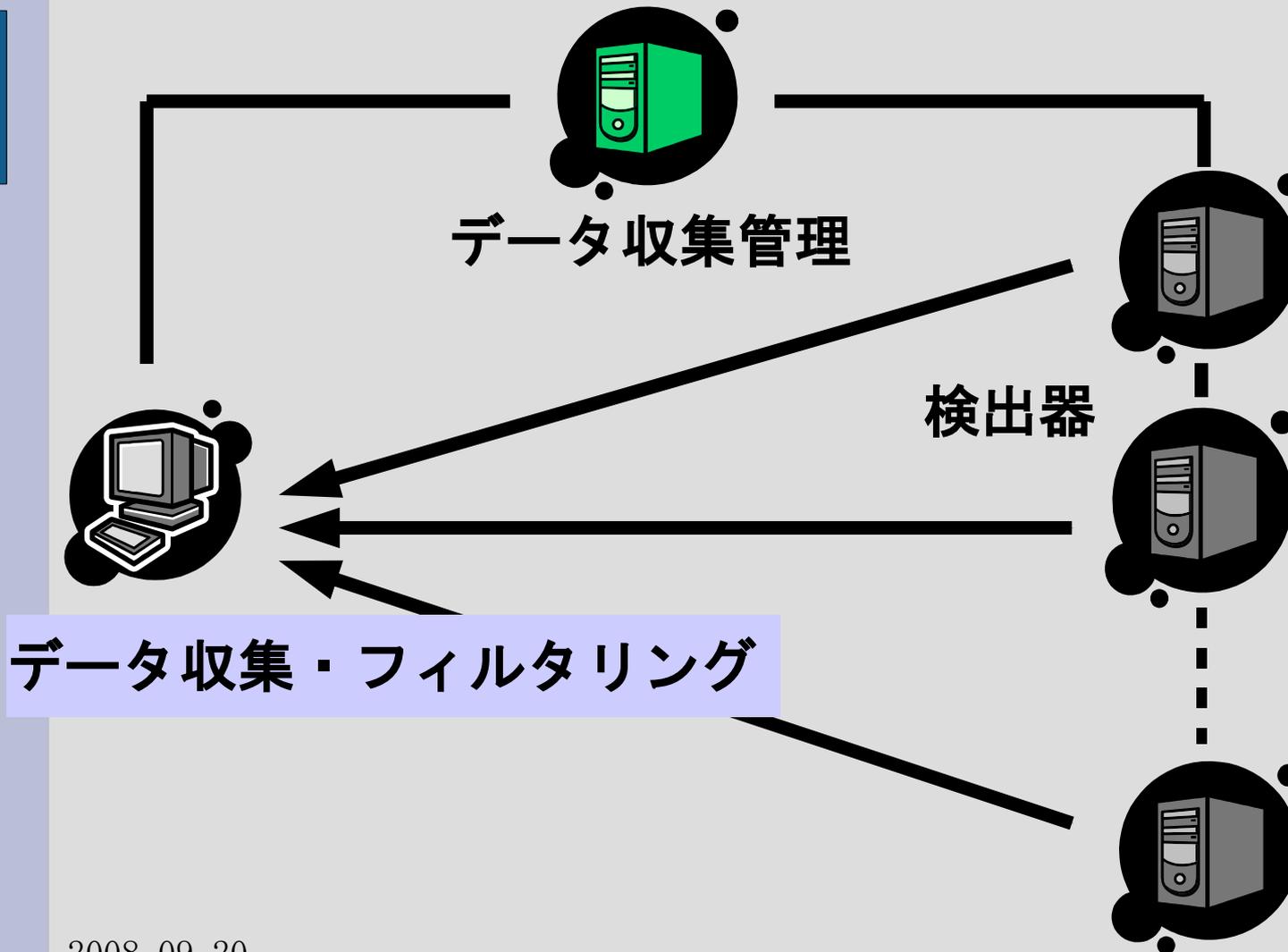
- ・ 個別の管理しかできない
- ・ どのノードで異常が発生しているかが分かりにくい

■ 全ノードの状態を同時に監視するための可視化システム

- ・ グラフや図形などを表示
- ・ システムの状態監視、分析が容易

TDAQ システム

■ TDAQ システムの概略



■ サブシステム

- Data Flow
- HLT
- Onlin Software
- DCS

TDAQ システム

■ 4つの主要なサブシステムで構成

(1) Data Flow システム

- 検出器からのデータを受信
- イベントビルダ
- HLT システムにデータのサブセットを供給
- 選択されたイベントデータを大容量記憶装置に転送

(2) HLT (High-Level Trigger) システム

- イベントの選択、フィルタリング、分類

TDAQ システム

■ 主要サブシステム

(3) Online Software システム

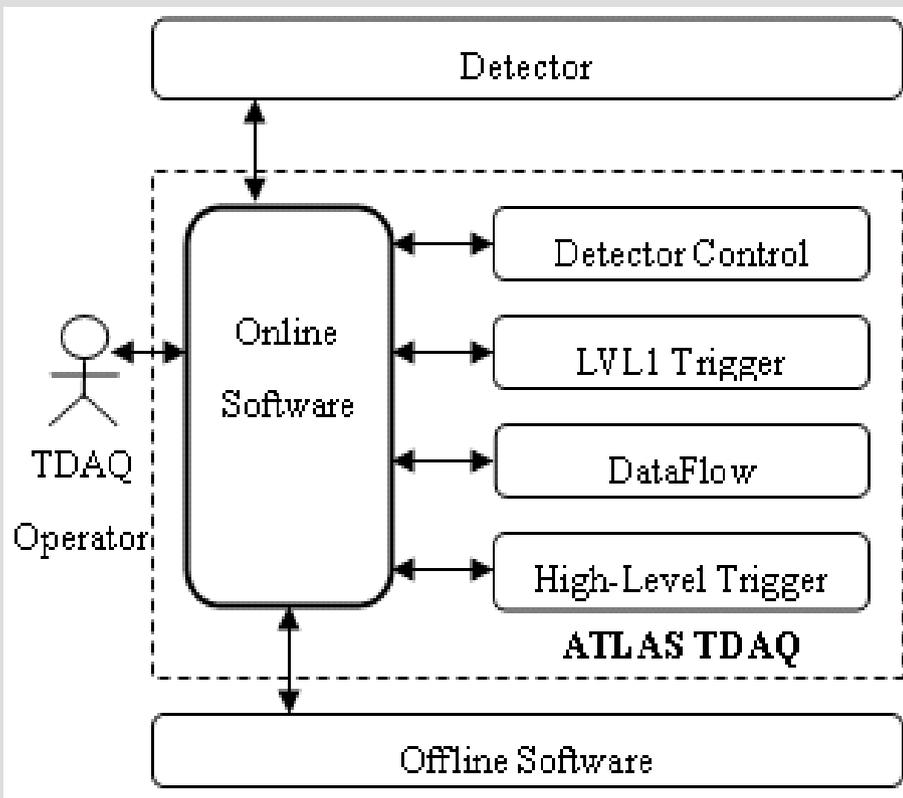
- ・ TDAQ システムの操作
- ・ データ取得の制御、テスト
- ・ 各ノードの状態監視

(4) DCS (Detector Control System)

- ・ ATLAS 検出器へ初期設定値の転送
- ・ ATLAS 検出器の操作

- ・ 各サブシステムは複数種類のノードで構成
- ・ ノードには管理・分析のためのパラメータが用意

Online Software システム

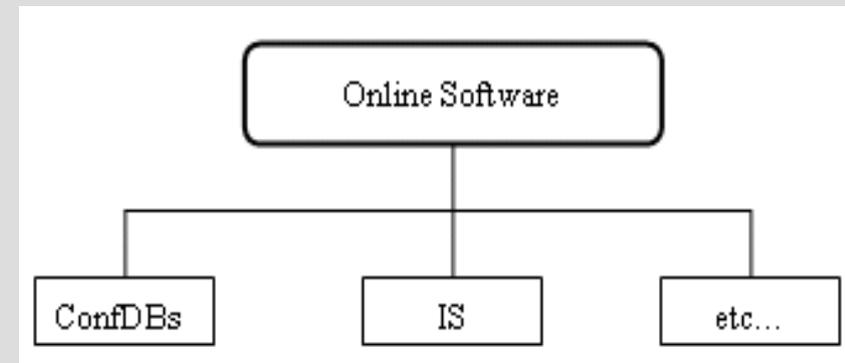


- ユーザインタフェースを提供
- 他のサブシステムと協調動作
- TDAQ システムの設定、制御、モニター機能
- オブジェクト指向技術により開発
- 変更可能なフレームワーク

Online Software フレームワーク

■ フレームワークの機能

- TDAQ システムの制御
- サブシステムの構成情報の保持
- アプリケーション間の情報共有



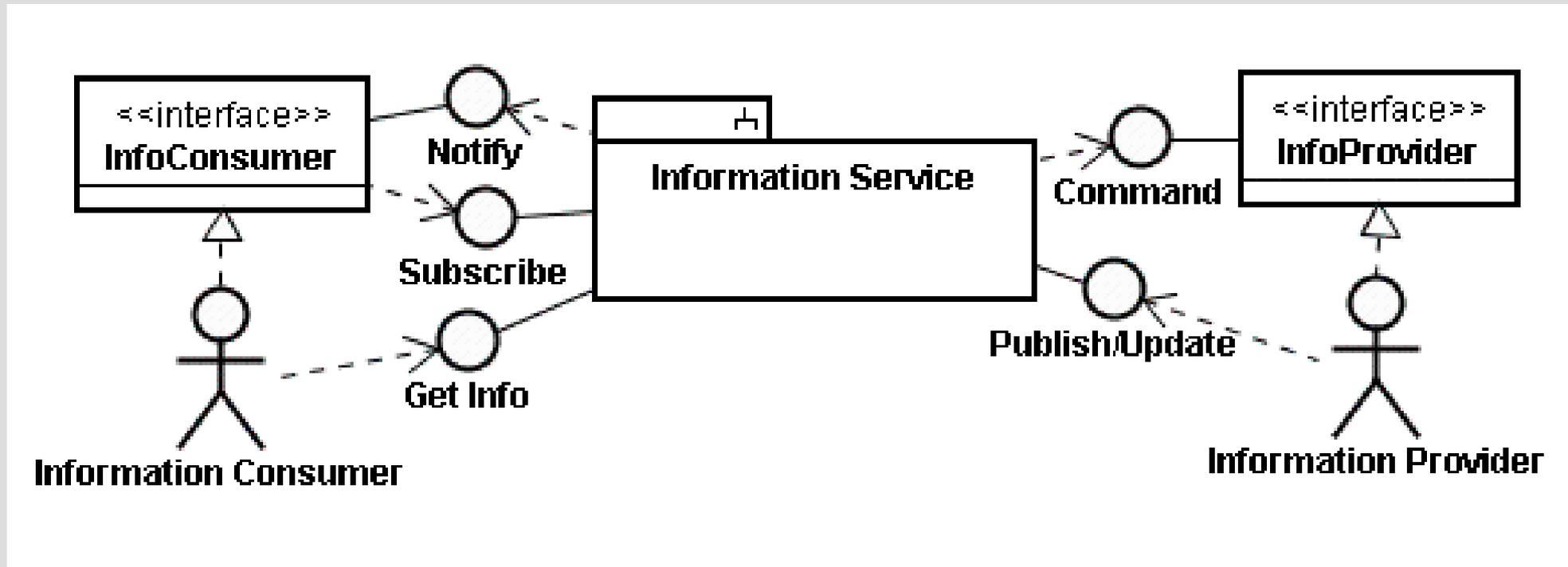
■ ConfDBs (Configuration Databases)

- ユーザーインタフェース
- アプリケーションプログラミングインタフェース
- 全てのサブシステムとノードの構成情報を保持
- ConfDBs からサブシステムやノードの種類、数を取得

Online Software フレームワーク

■ IS (Information Service)

- ・アプリケーション間での情報共有を提供



IGUI

■ IGUI : Integrated Graphical User Interface

- Online Software の GUI
- TDAQ システムを制御
- Online Software フレームワークと通信
- TDAQ システムの状態やパフォーマンスのモニタリング

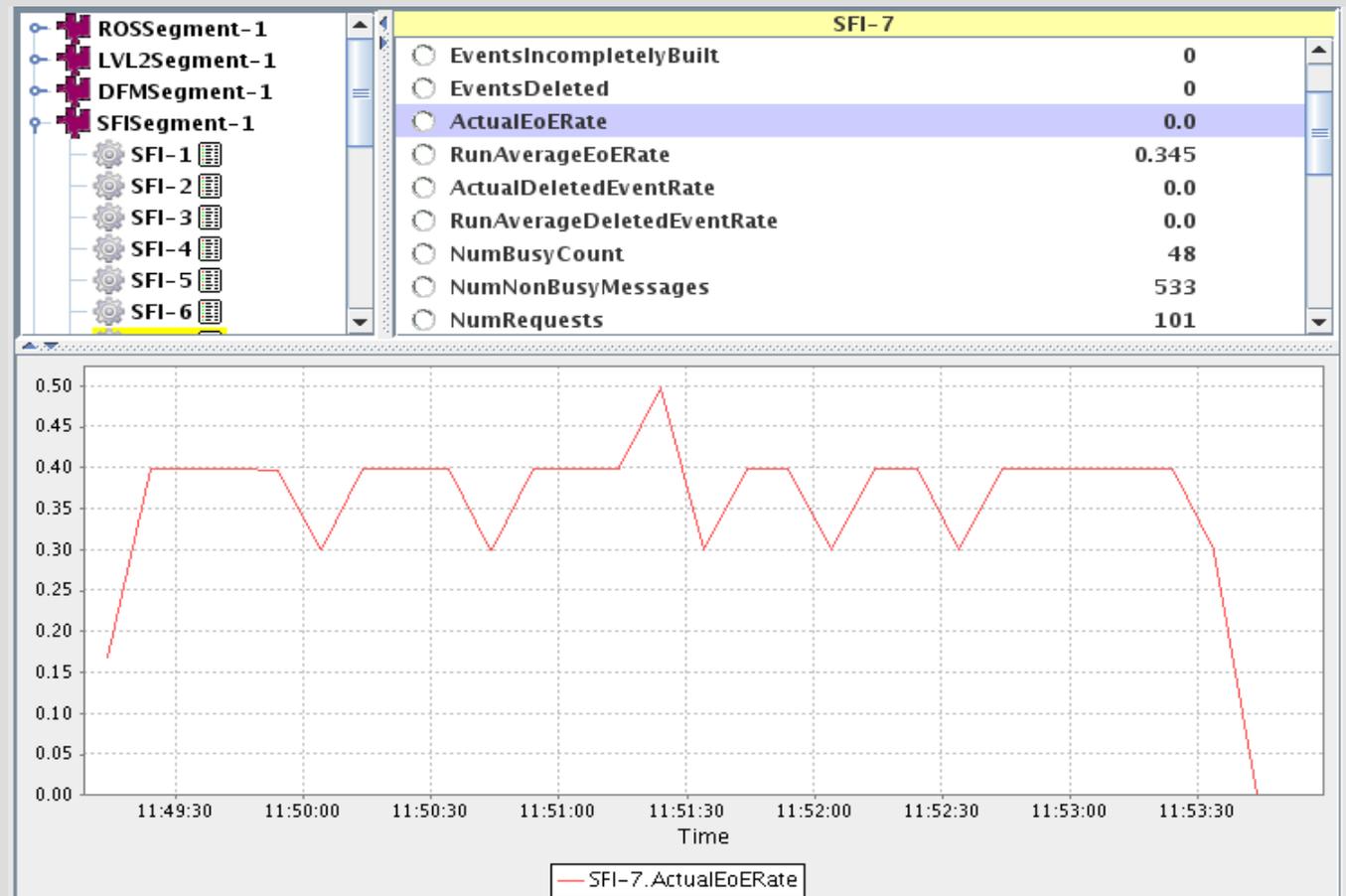
The screenshot shows the ATLAS TDAQ Software Graphical User Interface - Expert Control window. The interface is divided into several panels:

- Run control:** Contains buttons for RUN CONTROL STATE (NONE), Shutdown, and Boot. Below are buttons for Terminate, Unconfig, Stop, Pause, Initialize, Config, Start, and Continue.
- Run Information:** Displays run details such as Run type (Physics), Run number (1220075200), Lumi block, Recording (Disable), Run Start Time (30/08/08 14:46:40), Run Stop Time (30/08/08 14:51:30), Total run time (00:04:50), and a table for Level 1, Level 2, Event Builder, Event Filter, and Recorded Events.
- Monitor:** Shows a tree view of system components, including RootController, ROSegment-1:dhcp-196, LVL2Segment-1, DFMSegment-1:dhcp-196, SFISegment-1:dhcp-196, and EventFilter:dhcp-196-209.
- RootController Panel:** Displays application status (UP), Run Control State (NONE), Busy Status (FREE), Fault Status (OK), and Command (NONE). It also includes Operational Monitoring (publish state, publish statistics), Debug Level Control (0, Set debug level), Membership (IN, OUT), and Recovery commands (retry, ignore error, restart, kill).

IGUI

■ Data Flow Panel

- ConfDBs や IS からノードの構成情報やパラメータを取得
- グラフ表示



イベントビルダモニタリングシステム

■ 既存の IGUI

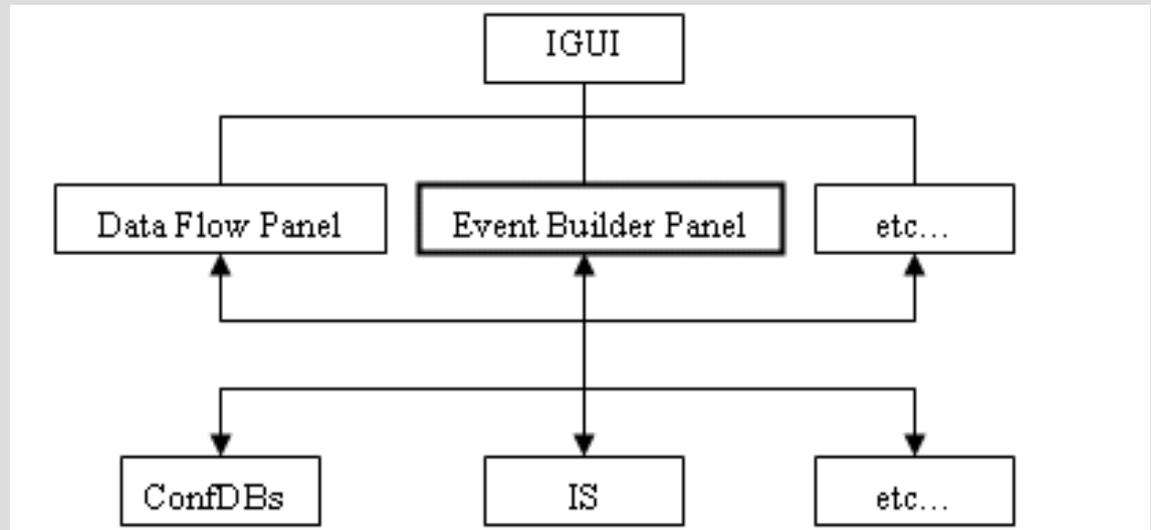
- ・ 一つ一つのノードのパラメータを取得、表示
- ・ ノードは数百台、数千台に及ぶ
- ・ 全てのノードを同時に表示させることができない
→ システム全体の状態監視、分析が困難

■ Event Builder Panel

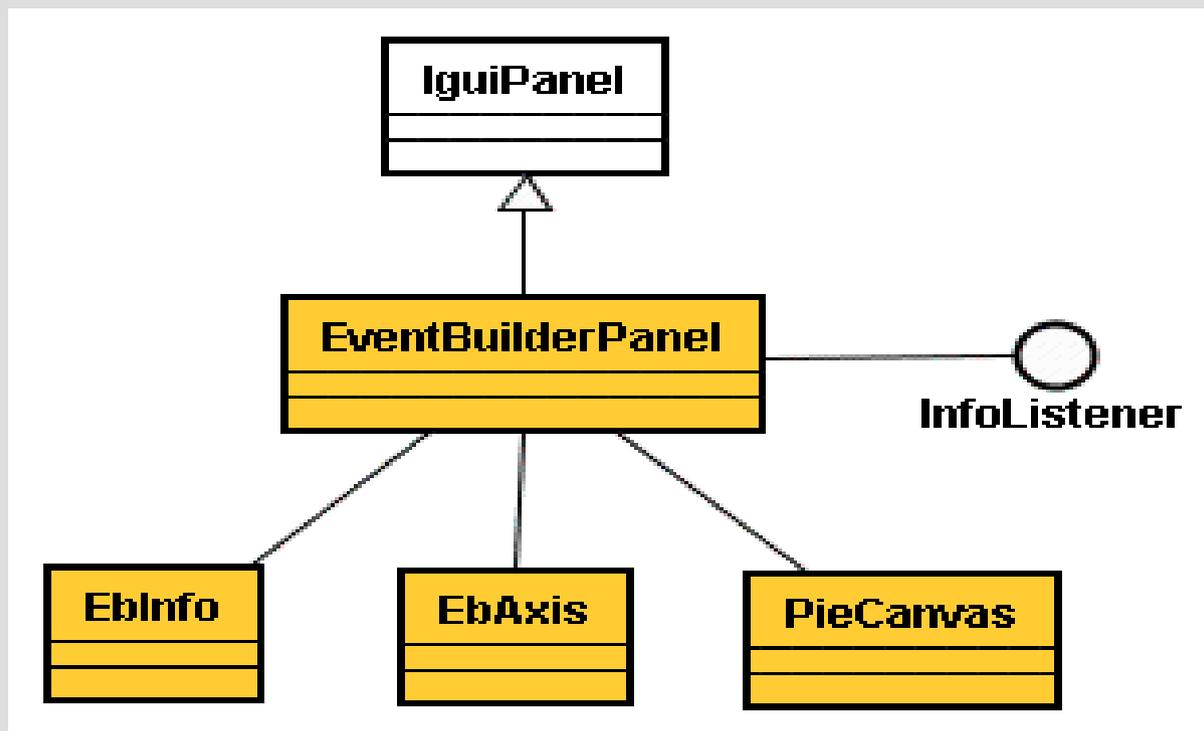
- ・ 全てのノードの状態を同時に表示
- ・ システム全体の状態監視、分析が容易
- ・ 異常をきたしている部分を簡単に発見
- ・ ノードの管理が容易

Event Builder Panel

- Java 言語を用いたオブジェクト指向技術によって開発
- Online Software フレームワークと協調動作
- ConfDBs からシステム全体の構成情報を取得
- IS から全てのノードのパラメータを動的に取得
- サブシステムの種類ごとにノードを整理
- 同一種類の全てのノードのパラメータを同時に表示
- 既存の IGUI のアプリケーションの一つとして動作



Event Builder Panel : クラス図



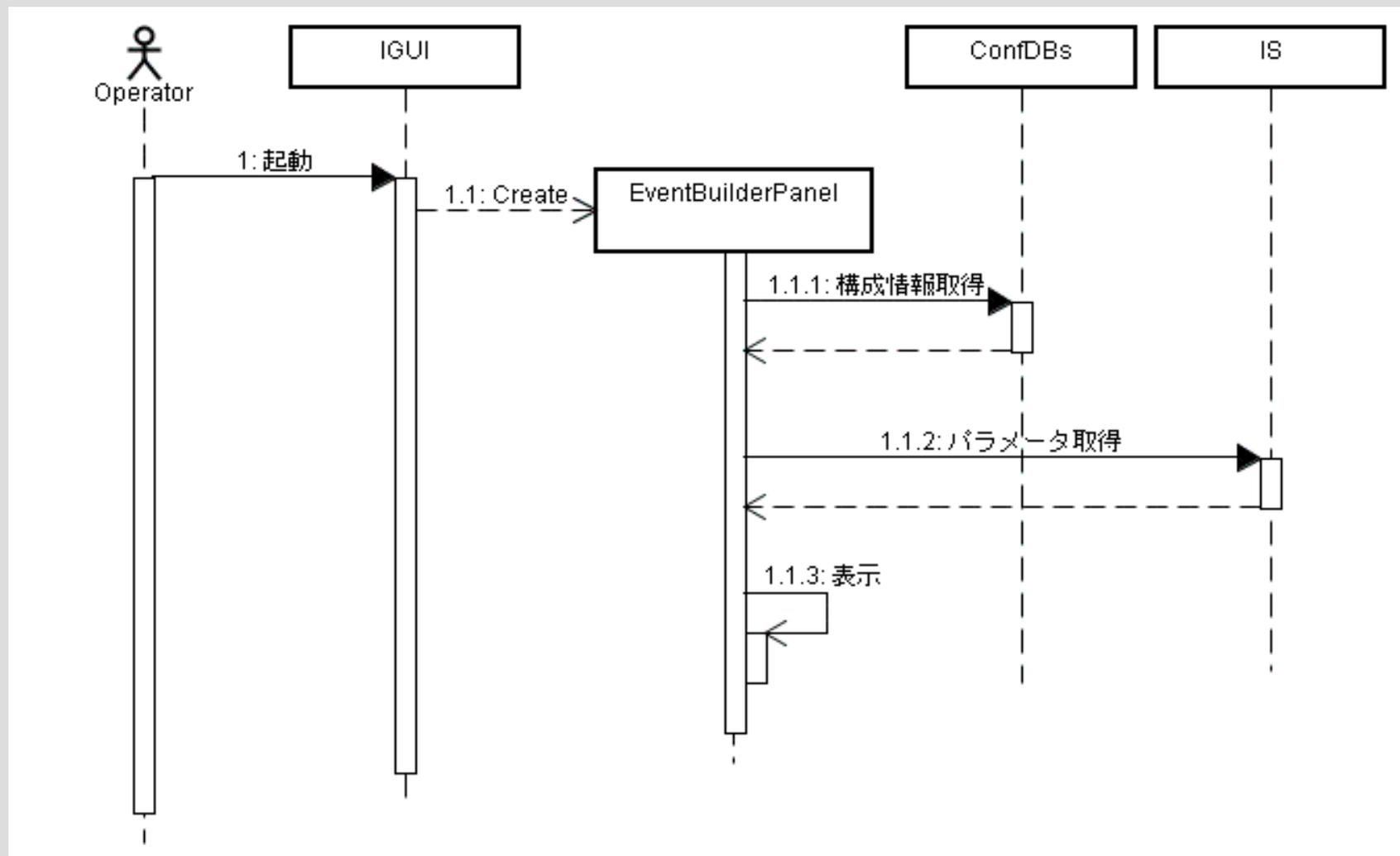
■ IguiPanel

- ・ IGUI パッケージのクラス

■ InfoListener

- ・ IS と通信するためのインタフェース

Event Builder Panel : シーケンス図



■ ConfDBs、ISからシステムの構成情報、ノードのパラメータ取得

Event Builder Panel : 機能

■ グラフ表示

- ・ 各ノードのパラメータの値を取得
- ・ 棒グラフとヒストグラムとして表示
- ・ グラフ上でマウスをドラッグするとグラフが拡大
- ・ 拡大処理は棒グラフとヒストグラムそれぞれ相互に対応

■ ノード状態表示

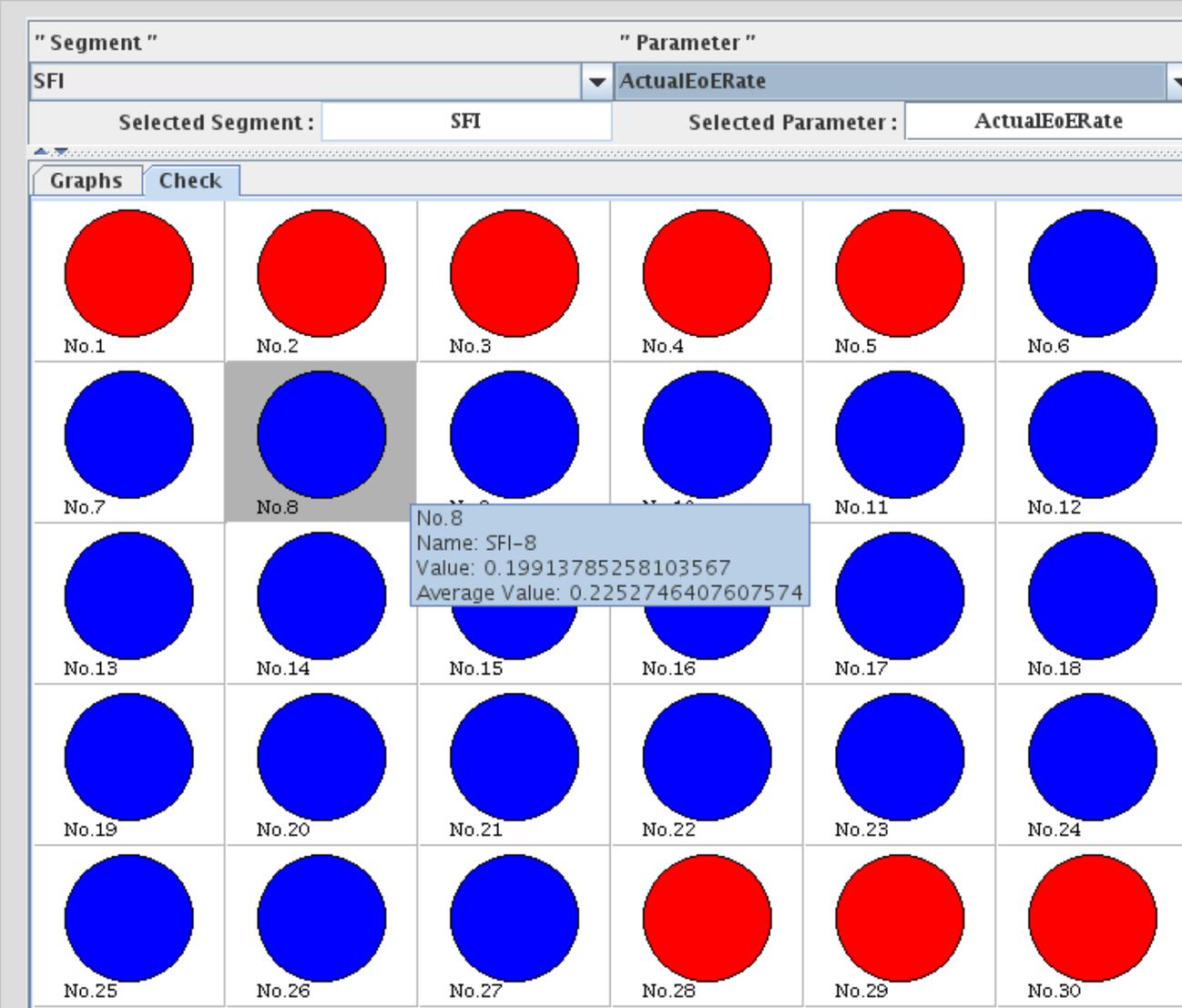
- ・ 各ノードを円で表示
 - ・ ノードのパラメータの値によって円の色を変更
-
- ・ データ収集システム全体の状態を可視化

シミュレーション

■ TDAQ のシミュレーション

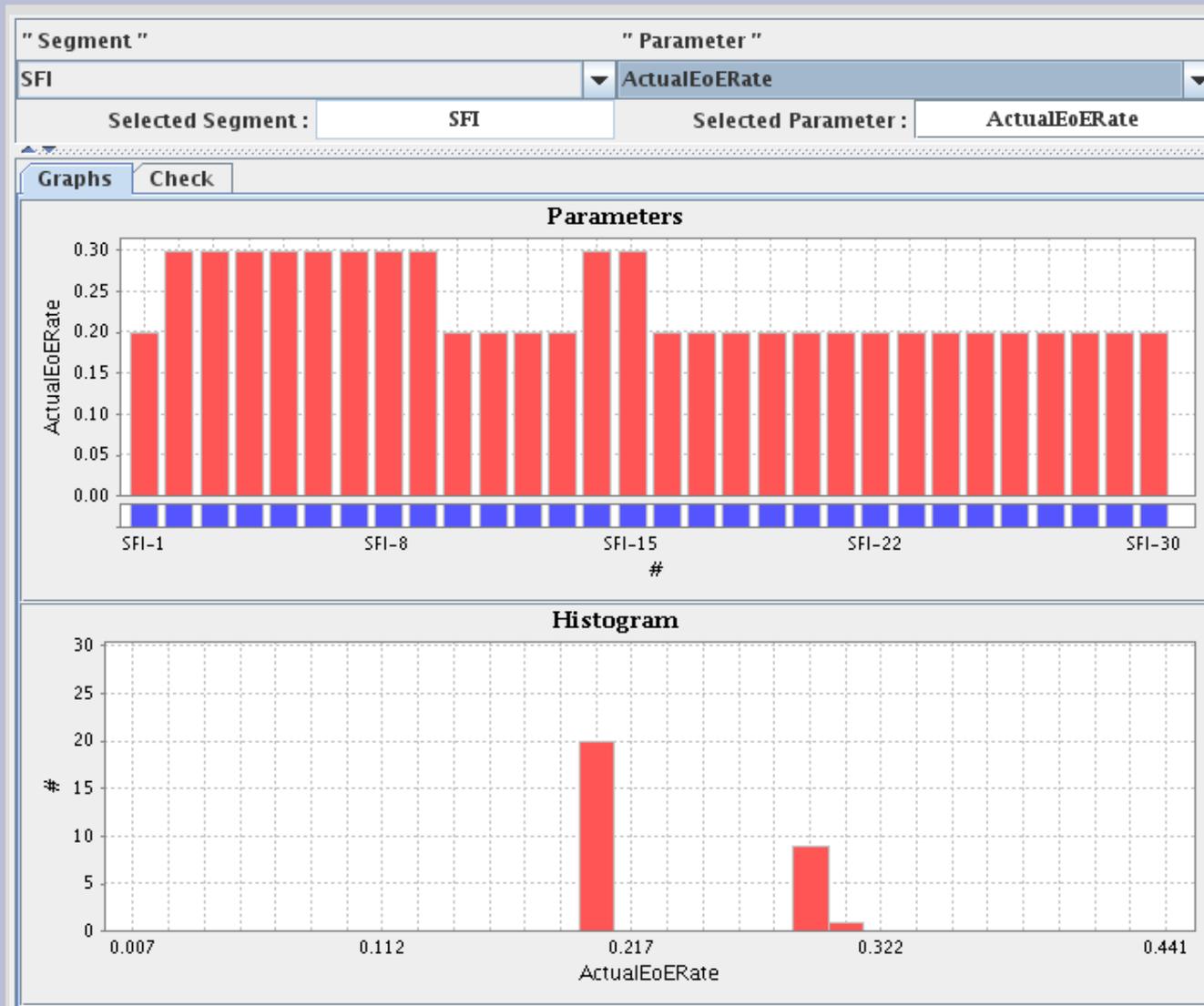
- Event Builder Panel を
実装した IGUI でシミュレーション
- 1 台の PC 内で TDAQ システムを仮想的に動作
- ノード数 : 30 台
- OS : SLC4
- TDAQ システムのバージョン : tdaq-01-09-00

Event Builder Panel : 表示①



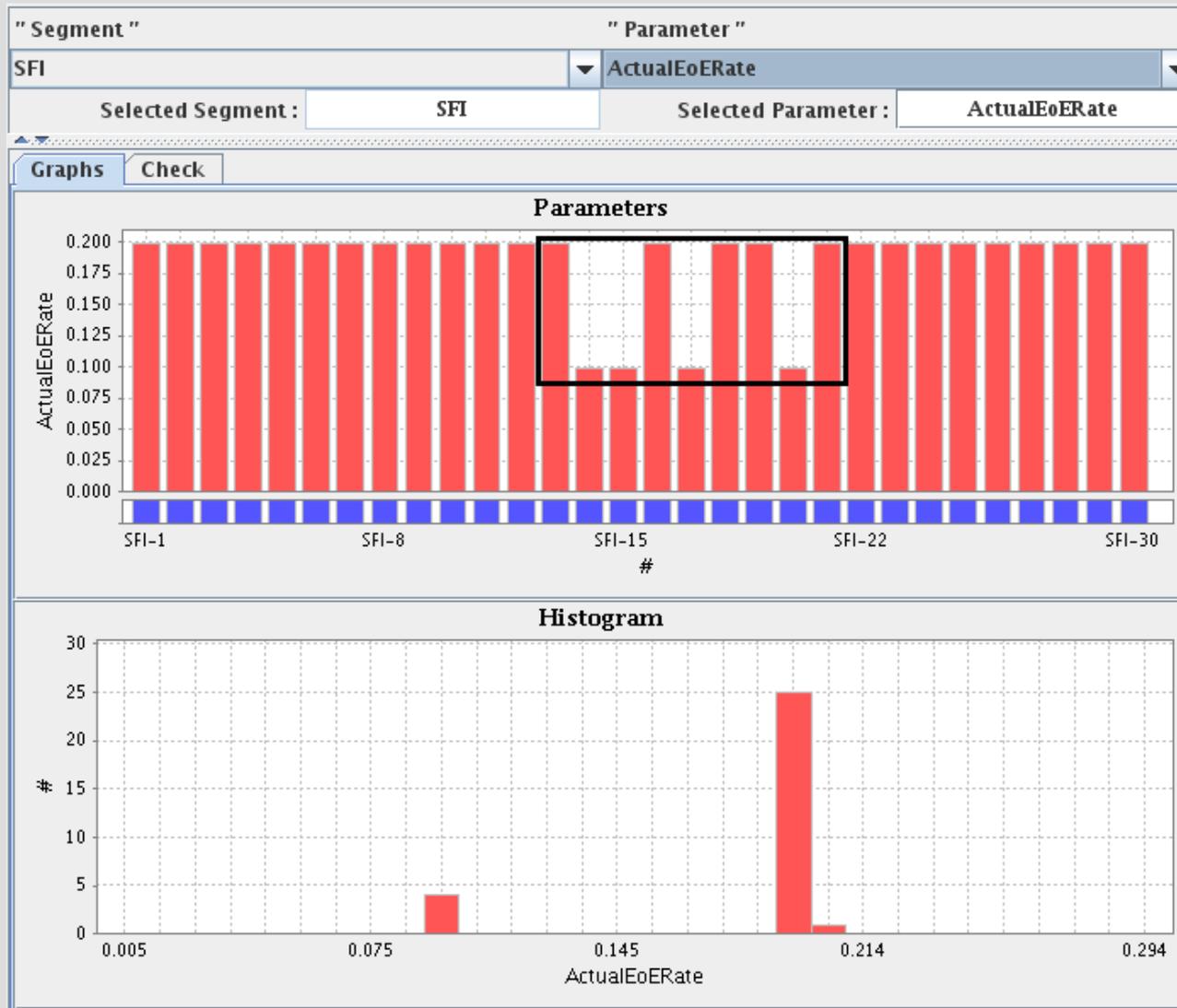
- 各ノードの状態を簡易的な図で表示
 - パラメータの値によって色を変更
 - 正常値：青色
 - 異常値：赤色
-
- どのノードに異常があるか

Event Builder Panel : 表示②

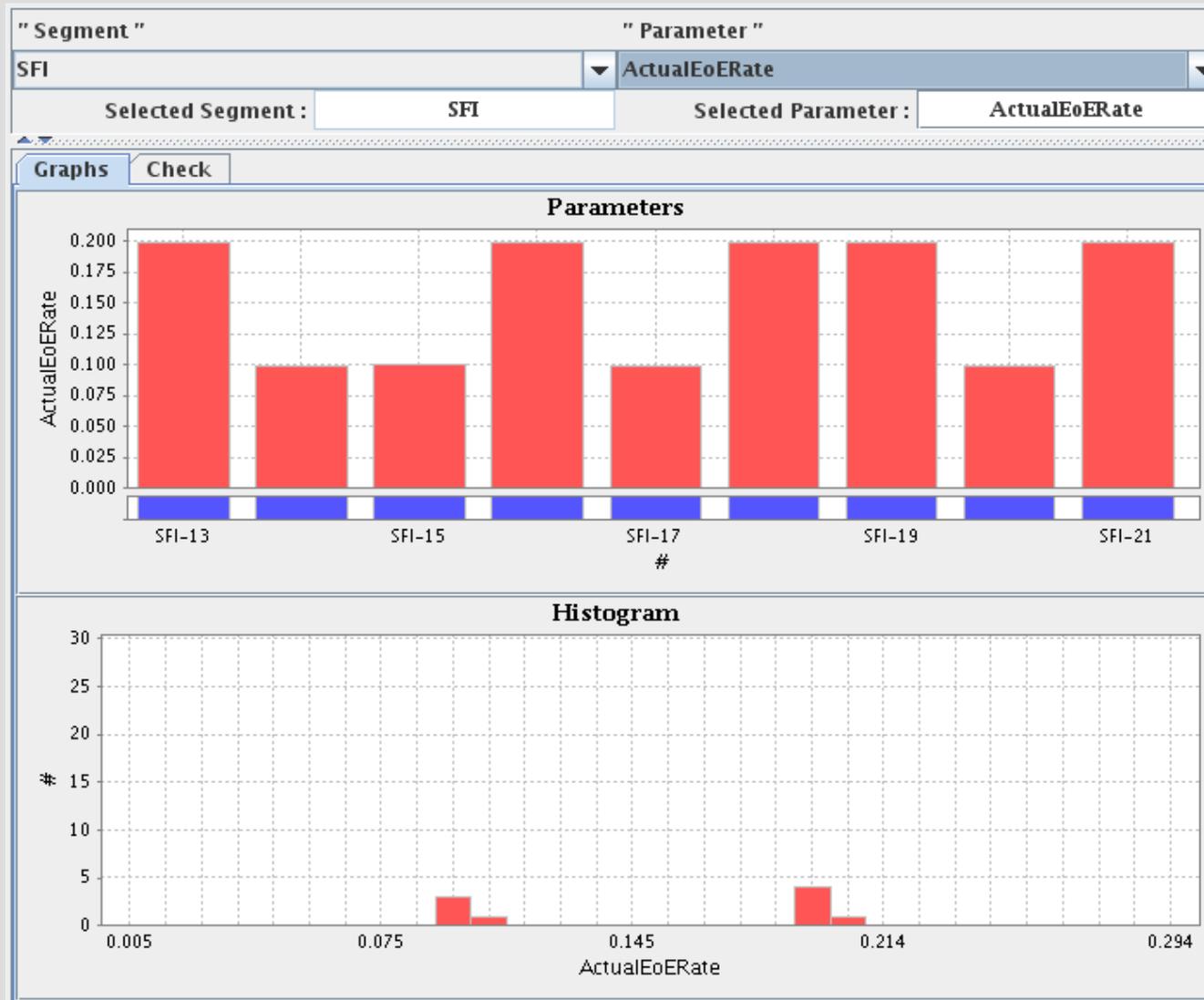


- ・ パラメータの値を棒グラフとヒストグラムで表示
- ・ グラフ上でドラッグすると拡大処理
- ・ 各ノードのパラメータの値の比較
- ・ パラメータの分布

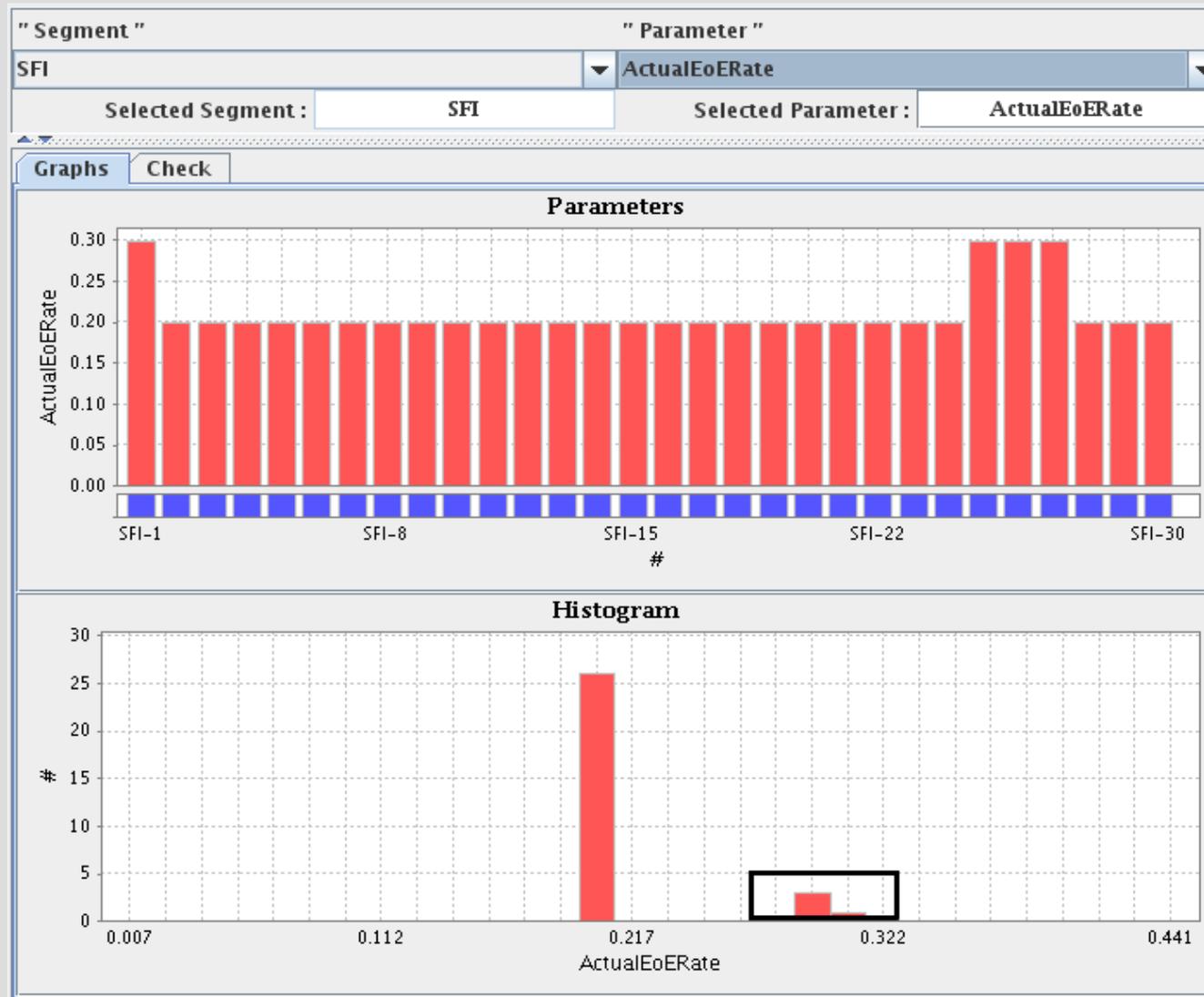
Event Builder Panel : グラフ操作①



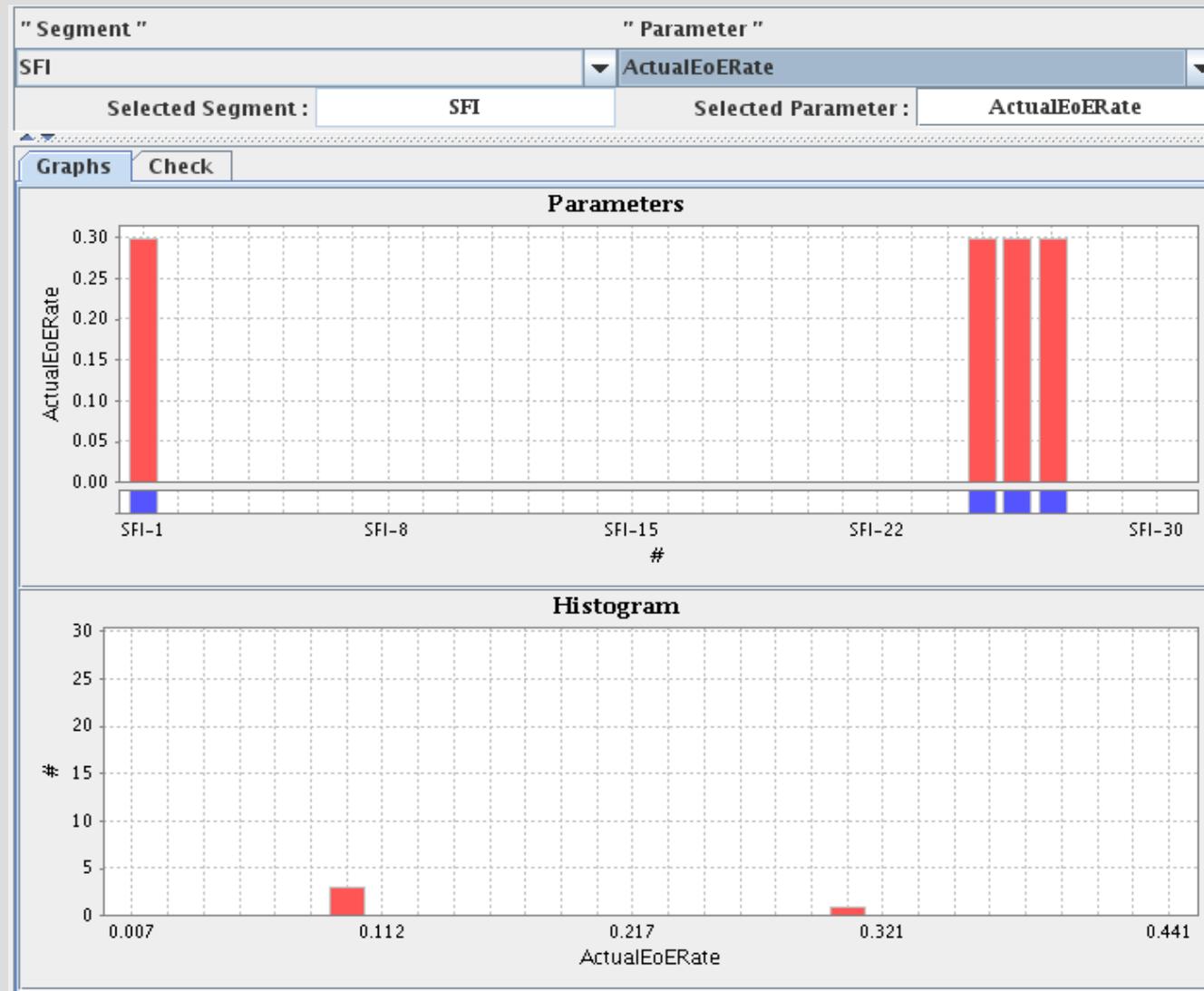
Event Builder Panel : グラフ操作②



Event Builder Panel : グラフ操作③



Event Builder Panel : グラフ操作④



まとめ

■ Event Builder Panel

- ・ データ収集システムの状態を可視化
- ・ 同時に全てのノードの状態を表示
- ・ 異常な部分を迅速に発見
- ・ システムの状態監視、分析が容易

■ 今後の展望

- ・ より分かりやすい表示
- ・ 実際の実験への適応