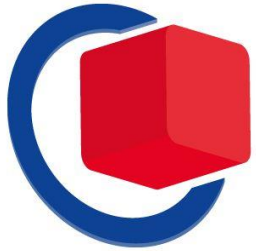


embeddedworld
Exhibition&Conference

Clock Manager: Revolutionizing Industrial Clock Synchronization with Real-Time Monitoring

Presenter: Voon, Weifeng (weifeng.voon@intel.com)
Intel Corporation, Penang, Malaysia

Authors: Lai, Jun Ann (peter.jun.ann.lai@intel.com)
Song, Yoong Siang (yoong.siang.song@intel.com)
Ahmad Tarmizi, Noor Azura (noor.azura.ahmad.tarmizi@intel.com)
Voon, Weifeng (weifeng.voon@intel.com)

intel

Agenda

- Introduction
- Software Architecture
- Functionality and Operational Flow
- Test Results
- Conclusion and Future Work

Introduction

Time-Sensitive Networking (TSN):	• Revolutionizes industrial automation, automotive systems, and real-time applications by ensuring deterministic and reliable communication.
Precision Time Protocol (PTP):	• Key to achieving high-precision time synchronization in TSN networks, surpassing Network Time Protocol (NTP) and GPS in accuracy.
Challenges in Time Sync:	• Network load variations, hardware imperfections, and environmental factors can cause synchronization errors, leading to data loss, increased latency, and system failures.
Out-of-Sync Impact:	• Out-of-sync devices can disrupt operations in industrial automation and automotive networks, posing safety risks.
SW Licensing Issues:	• Proprietary applications face challenges in accessing synchronization status due to non-permissive licenses of time synchronization daemons.
Clock Manager Solution:	• Introduces a time monitoring solution for real-time status updates from synchronization daemons, enhancing functionality, security, and safety without exposing proprietary code.

Clock Manager

Overview:

- Open-source C++ tool for monitoring clock synchronization in industrial settings.
- Provides real-time synchronization status to subscribed user applications.

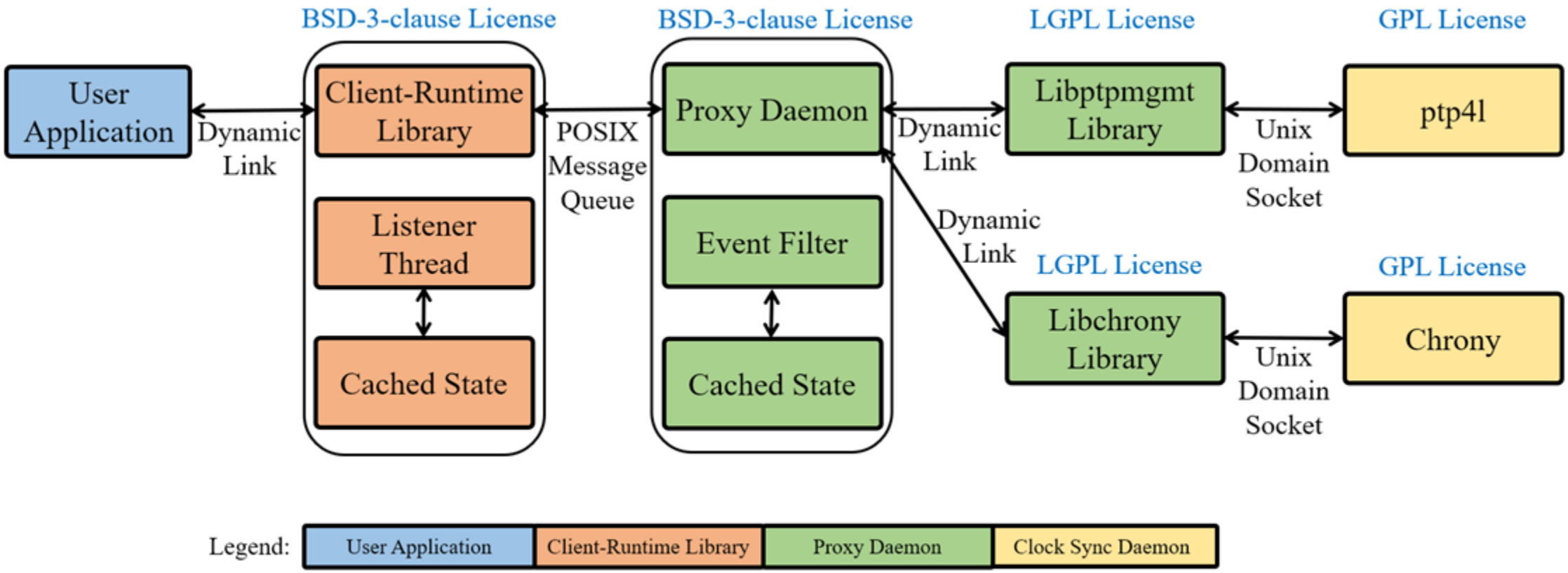
Architecture Components:

- **Proxy Daemon:**
  - Interfaces with ptp4l and Chrony to obtain real-time clock sync status.
  - Uses libptpmgmt for ptp4l and libchrony for Chrony via Unix Domain Sockets (UDS).
  - Filters and extracts synchronization data, passing it to the client-runtime library via message queue IPC.
- **Client-Runtime Library:**
  - Offers an API for user applications to connect and receive synchronization events.
  - Stores real-time status and detects changes, triggering event notifications.

Benefits:

- Reduce complexity of handling clock sync status in a single proxy and free the TSN applications from that complexity.
- Proprietary-friendly solution for seamless integration.

High-Level Architectural Overview



Functionality and Operational Flow

Main Function:

- Interfaces with clock sync daemons to provide real-time telemetry data for user applications.
- Supports detection of synchronization errors.

Operational Flow:

- Execute ptp4l and Chrony, then run the proxy daemon.
- User applications connect to the proxy daemon via the CONNECT API, receiving a session ID.
- Subscribe to events using the SUBSCRIBE API, specifying interest in event types.
- Monitor data changes with the WAIT API, which blocks until event changes occur or a timeout elapses.
- Immediate notifications for out-of-sync events enable swift reactions to prevent machine damage.

Functionality and Operational Flow

ptp4l Supported Events:

- **offset_in_range:** Indicates if the offset between Grandmaster (GM) and PHC is within limits; provides latest clock offset.
- **synced_to_primary_clock:** Indicates GM identification and synchronization status.
- **as_capable:** Checks if the network link partner supports IEEE 802.1AS timing requirements.
- **gm_Changed:** Indicates change in GM candidate with UUID information.

Chrony Supported Events:

- **clock_offset:** Indicates if the offset between reference and system clock is within limits; provides latest clock offset.
- **clock_reference_id:** Provides ID of the reference clock source.
- **polling_interval:** Indicates Chrony's synchronization period in nanoseconds.

Result: Overview of Data Obtained

Provides a comprehensive view of synchronization status and event timing.

Enhances user applications' ability to make informed decisions based on the most current data.

Data Obtained from Clock Manager

```
[clkmgr] Connected. Session ID : 0
[clkmgr] set subscribe event : 0xf
[clkmgr] set composite event : 0x7
GM Offset upper limit: 100 ns
GM Offset lower limit: -100 ns
Chrony Offset upper limit: 100 ns
Chrony Offset lower limit: -100 ns

[clkmgr][1315.907] Obtained data from Subscription Event:
+---------------------------+-----------------------+
| Event                     | Event Status          |
+---------------------------+-----------------------+
| offset_in_range           | 1                     |
| synced_to_primary_clock   | 1                     |
| as_capable                | 1                     |
| gm_Changed                | 1                     |
+---------------------------+-----------------------+
| GM UUID                   | 222211.fffe.011122    |
| clock_offset              | 2                  ns |
| notification_timestamp    | 1925965364441672526 ns |
+---------------------------+-----------------------+

+---------------------------+-----------------------+
| chrony offset_in_range    | 1                     |
+---------------------------+-----------------------+
| chrony clock_offset       | 0                  ns |
| chrony clock_reference_id | 50484330              |
| chrony polling interval   | 500000             us |
+---------------------------+-----------------------+
```

Result: GM Event Notification

Clock Synchronization Process:

- **gm_Changed Event**:
  - Clock Manager notifies the sample application and provides the UUID of the new GM.
  - Set to TRUE, indicating there is new GM.
- **offset_in_range Event**:
  - When ptp4l starts synchronizing the PHC, the clock offset is initially large [> +-100].
  - Set to FALSE, indicating clock offset is out-of-range.
- **synced_to_primary_clock Event**:
  - ptp4l status changes from uncalibrated to time receiver (slave).
  - Set to TRUE, indicating successful synchronization with the GM.

Significance:

- Highlights the role of Clock Manager in providing timely updates to applications, ensuring accurate time coordination

Data Obtained from Clock Manager	`ptp4l` log

```
[clkmgr][1329662.491] Obtained data from Notification Event:
+--------------------------+-------------+-------------+
| Event                    | Event Status | Event Count |
+--------------------------+-------------+-------------+
| offset_in_range          | 0           | 0           |
| synced_to_primary_clock  | 1           | 1           |
| as_capable               | 1           | 0           |
| gm_Changed               | 1           | 1           |
+--------------------------+-------------+-------------+
| GM UUID                  |    222211.fffe.011122     |
| clock_offset             |    1087                ns |
| notification_timestamp   |    1930695793739487850 ns |
+--------------------------+-------------+-------------+
```

```
ptp4l[1329662.355]: selected best master clock 222211.fffe.011122
ptp4l[1329662.355]: port 1 (enp1s0): LISTENING to UNCALIBRATED
_SLAVE
ptp4l[1329662.490]: master offset       1087 s2 freq   +2704 pa
lay         9
ptp4l[1329662.490]: port 1 (enp1s0): UNCALIBRATED to SLAVE on M
_CLOCK_SELECTED
```

Result: Clock Offset Notification

Event Notification:

- **offset_in_range Event:**
  - Triggered when the clock offset falls within the predefined range [-100, 100].
  - Clock Manager immediately notifies the sample application that subscribed to this event.
  - Transition from out-of-range to in-range.

Significance:

- Demonstrates the Clock Manager's ability to monitor and report changes in synchronization status.
- Ensures applications are informed of stable synchronization, enhancing system reliability and performance.

Data Obtained from Clock Manager

```
[clkmgr][1677.163] Obtained data from Notification Event:
+--------------------------+--------------+-------------+
| Event                    | Event Status | Event Count |
+--------------------------+--------------+-------------+
| offset_in_range          | 1            | 1           |
| synced_to_primary_clock  | 1            | 0           |
| as_capable               | 1            | 0           |
| gm_Changed               | 0            | 0           |
+--------------------------+--------------+-------------+
| GM UUID                  |     222211.fffe.011122     |
| clock_offset             |     -99                 ns |
| notification_timestamp   |     1925965725696221160 ns |
+--------------------------+--------------+-------------+
```

ptp4l log

```
ath delay         8
ptp4l[1677.144]: master offset       -113 s2 freq   -1092 p
ath delay         8
ptp4l[1677.152]: master offset       -111 s2 freq   -1091 p
ath delay         8
ptp4l[1677.160]: master offset        -99 s2 freq   -1059 p
ath delay         8
```

Result: Loss of GM Scenario

Test Overview:

- Simulated GM loss by terminating the GM, causing ptp4l to stop receiving periodic announce messages.
- Time synchronization halts on the local platform.
- Local PHC is temporarily assigned as the leader clock (master).

Clock Manager Response:

- Triggers the **synced_to_primary_clock** event to indicate the loss of GM

Significance:

- Demonstrates the Clock Manager's ability to detect and report changes in network synchronization status.
- Ensures users are informed of critical changes, allowing for appropriate responses to maintain network stability.

Data Obtained from Clock Manager

```
[clkmgr][1330491.559] Obtained data from Notification Event:
+--------------------------+-------------+-------------+
| Event                    | Event Status | Event Count |
+--------------------------+-------------+-------------+
| offset_in_range          | 1           | 1           |
| synced_to_primary_clock  | 0           | 1           |
| as_capable               | 0           | 0           |
| gm_Changed               | 1           | 1           |
+--------------------------+-------------+-------------+
| GM UUID                  |    00a0c9.fffe.000000    |
| clock_offset             |    0                  ns |
| notification_timestamp   |    1930696622803237820 ns |
+--------------------------+-------------+-------------+
```

ptp4l log

```
ptp4l[1330491.522]: master offset         -2 s2 freq    -602 pa
ptp4l[1330491.530]: master offset          0 s2 freq    -596 pa
ptp4l[1330491.553]: port 1 (enp1s0): SLAVE to LISTENING on ANNO
ptp4l[1330491.554]: selected local clock 00a0c9.fffe.000000 as
```

Result: Clock Jump Scenario

Experiment Overview:

- Simulated a loss of synchronization by advancing the GM clock to the year 2030, creating a significant time jump.

Clock Manager Response:

- Detected a substantial clock offset surge.
- Promptly alerted the user to the loss of synchronization, indicating the offset exceeded predefined limit

Performance Evaluation:

- Achieved a response time of approximately 7 milliseconds.
- Demonstrates the excellent responsiveness of the Clock Manager in detecting and reporting synchronization anomalies.

Data Obtained from Clock Manager

```
[clkmgr][1331333.530] Obtained data from Notification Event:
+--------------------------+--------------+-------------+
| Event                    | Event Status | Event Count |
+--------------------------+--------------+-------------+
| offset_in_range          | 0            | 1           |
| synced_to_primary_clock  | 1            | 0           |
| as_capable               | 1            | 0           |
| gm_Changed               | 0            | 0           |
+--------------------------+--------------+-------------+
| GM UUID                  |     222211.fffe.011122     |
| clock_offset             |     129954110462       ns  |
| notification_timestamp   |     1924876929958310090 ns |
+--------------------------+--------------+-------------+
```

ptp4l log

```
ptp4l[1331333.508]: master offset         -1 s2 freq    -609 pa
ptp4l[1331333.516]: master offset          1 s2 freq    -603 pa
ptp4l[1331333.523]: master offset 129954110462 s2 freq +6249999
ptp4l[1331333.531]: master offset 129953633056 s2 freq +6249999
```

Conclusion and Future Work

Clock Manager:

- Introduced as an innovative solution for monitoring and reporting clock synchronization status in industrial environments.
- Comprises a client-runtime library and a proxy daemon, enabling seamless integration with proprietary applications.
- Utilizes a permissive BSD-3-Clause license to ensure security and compliance with licensing requirements.

Functionality and Benefits:

- Communicates with time synchronization daemons like ptp4l and Chrony for real-time status updates.
- Notifies applications of synchronization events, such as GM status changes and clock offsets.
- Demonstrated rapid response time of approximately 7 milliseconds in detecting synchronization anomalies, outperforming ptp4l's synchronization interval.
- Provides a secure and efficient solution for managing clock synchronization in complex environments.

Future Work:

- Expand capabilities to support multiple PTP domains synchronization.

Thank You