JTRES 2014

Predictable Broadcasting of Parallel Intents in Real-Time Android

Igor Kalkov, Alexandru Gurghian, Stefan Kowalewski

Android & Real-Time

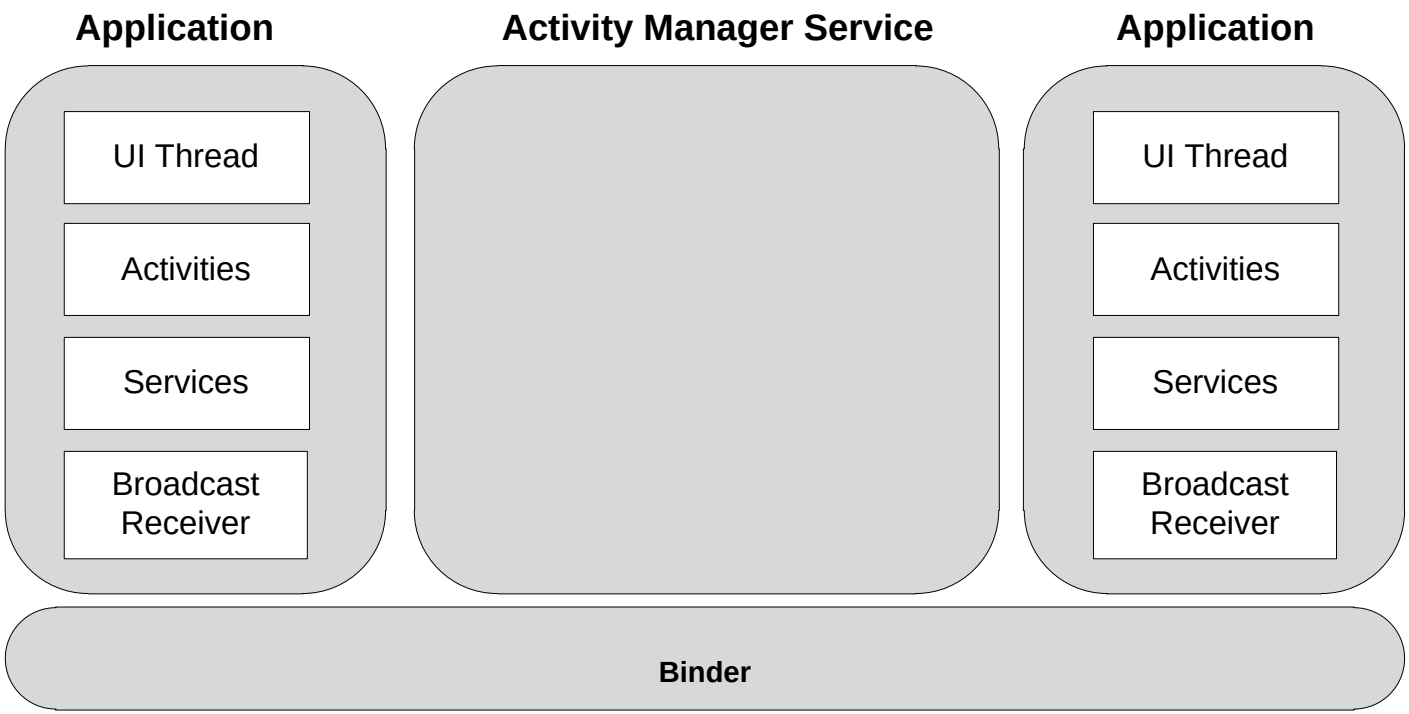
- A lot of research about real-time capabilities of Android
  - Performance evaluation [MH12; MN10; K13]
  - Critical components analysis [MM10; PF12]
- RTDroid [YK13]
  - RT Linux / RTEMS OS
  - Fiji real-time VM
- Real-time Extension for Android [KF12; GK13]
  - Kernel patched with RT_PREEMPT
  - Automatic real-time GC
  - Full backward compatibility

Android Application Components

- Android applications building blocks
  - Activities, Services, Content Providers, Broadcast Receivers
- Applications are maintained by own UI Threads
  - User input, lifetime-management,…
- Application linking by Intents
  - Inter- & intra-application communication
  - System event notifications
  - Broadcasts for generic message passing

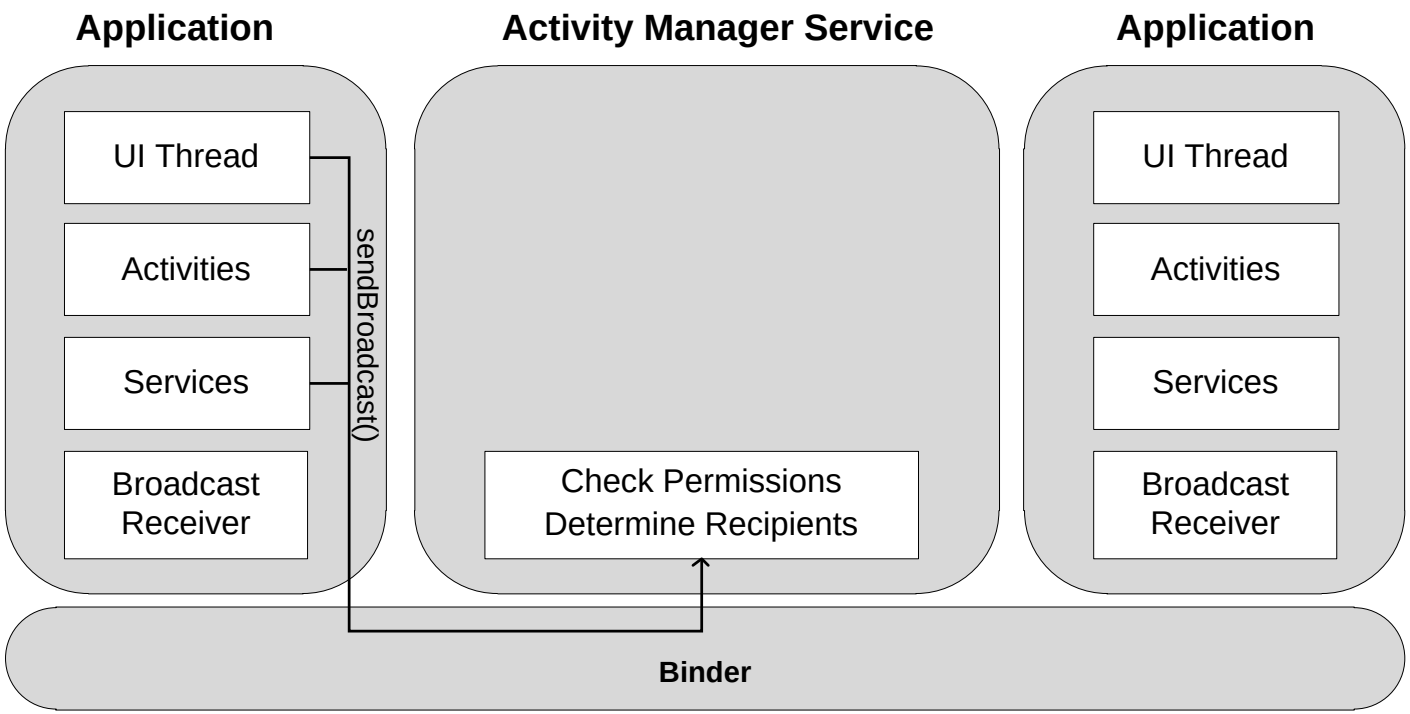
Inter-process Communication

- Application transmits a message across process borders



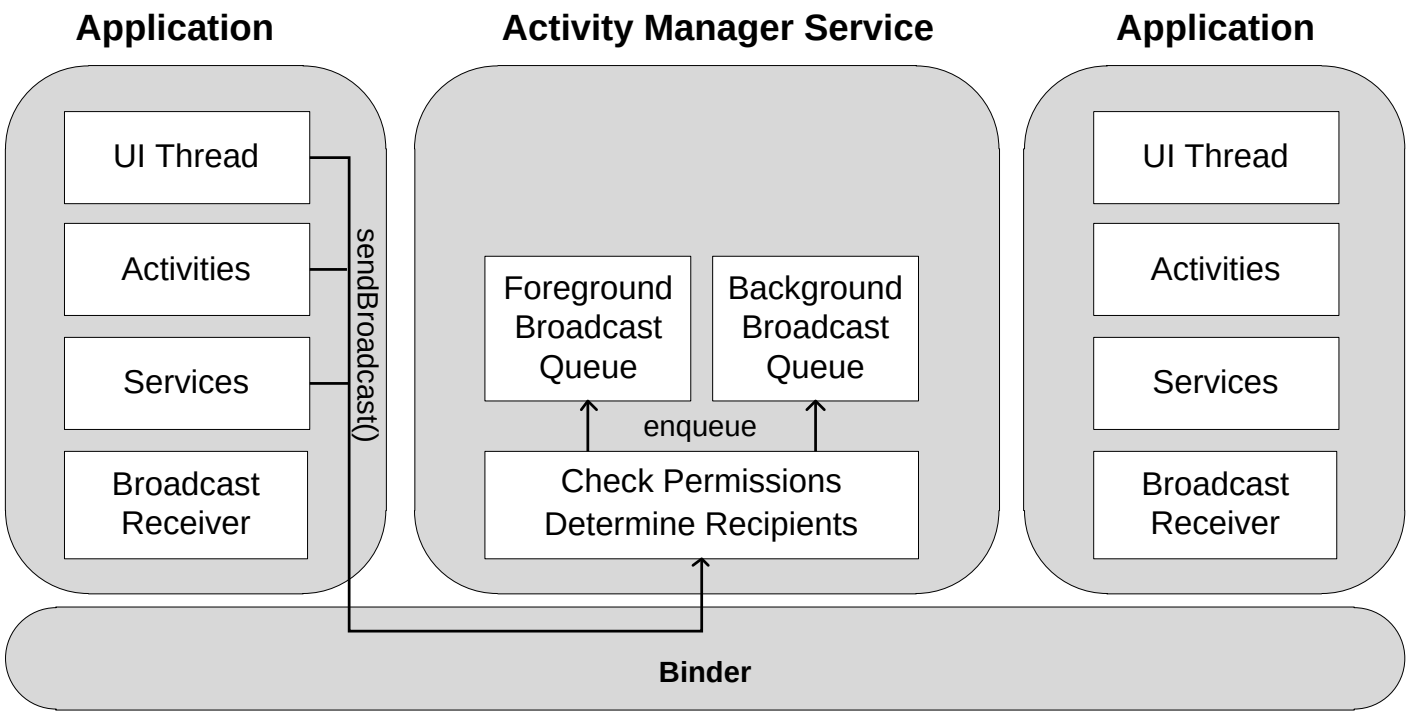
Inter-process Communication

- Application transmits a message across process borders
  1. Intent object is passed via sendBroadcast() to AMS



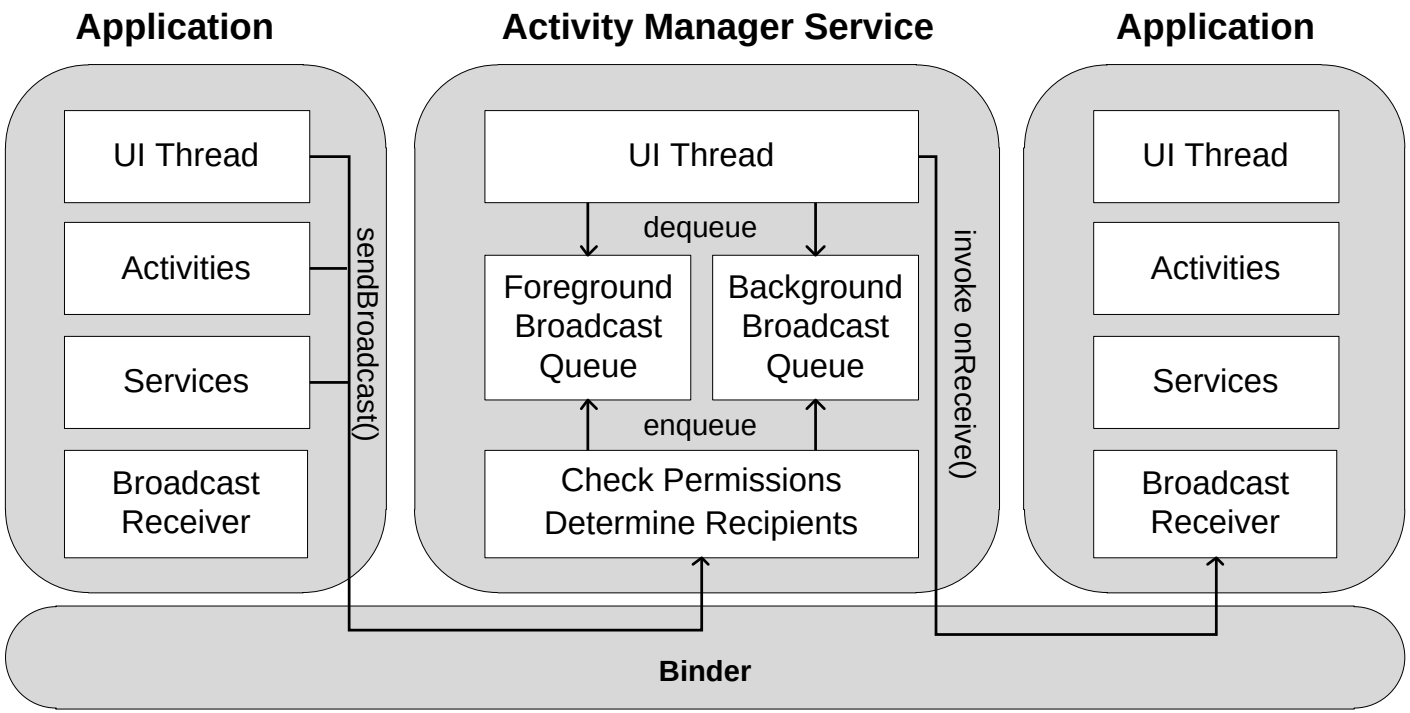
Inter-process Communication

- Application transmits a message across process borders
  1. Intent object is passed via sendBroadcast() to AMS
  2. Objects are queued depending on the target importance



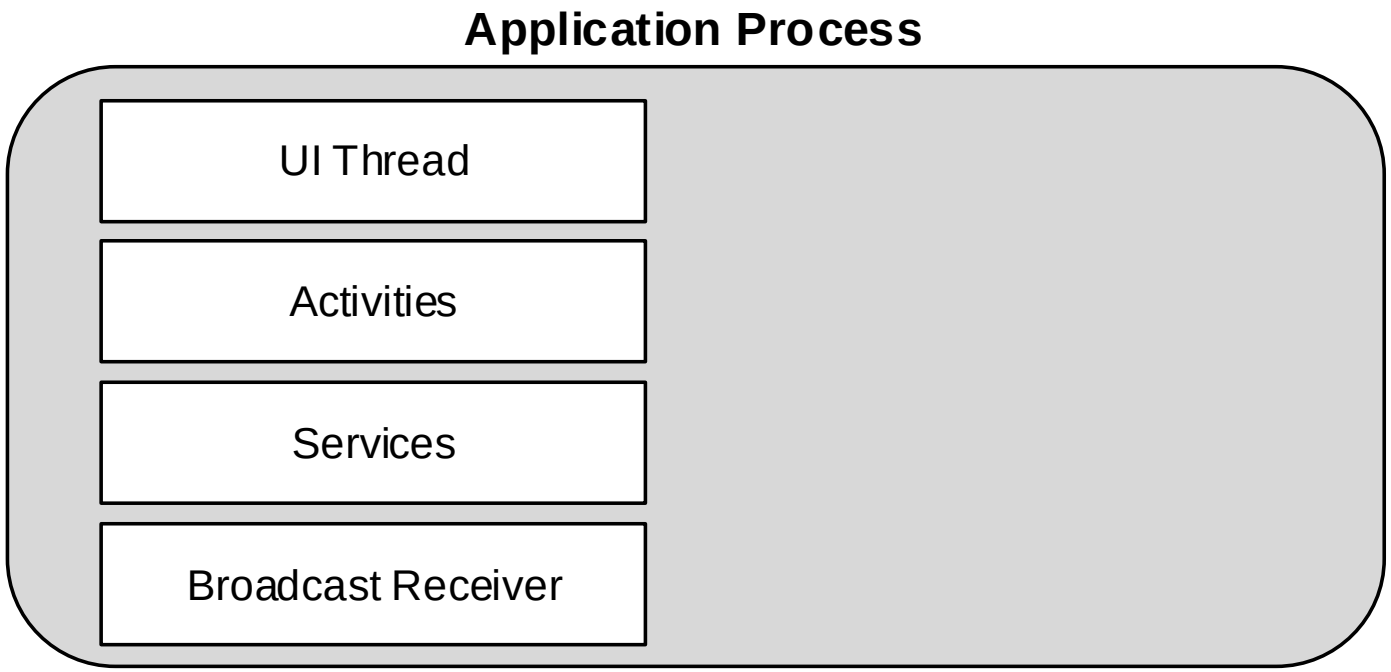
Inter-process Communication

- Application transmits a message across process borders
  1. Intent object is passed via sendBroadcast() to AMS
  2. Objects are queued depending on the target importance
  3. UI thread of the AMS transmits Intents to recipient application



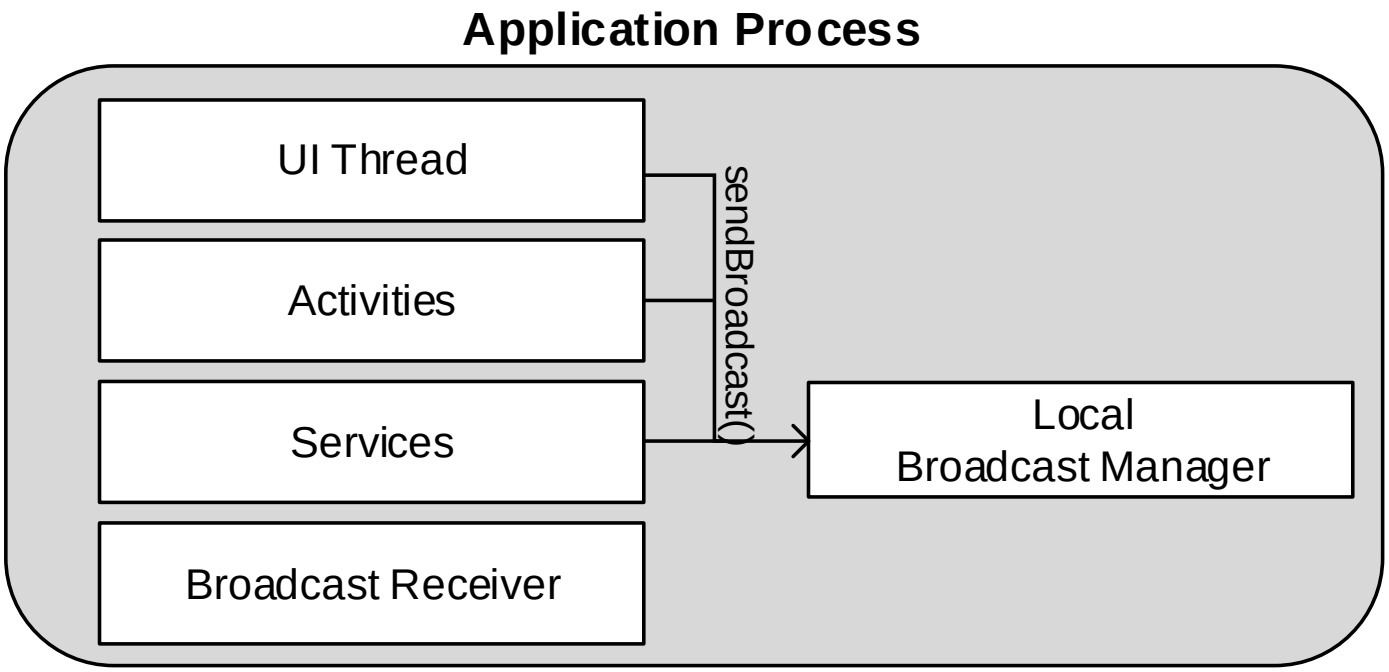
Asynchronous Local Messaging

- Intent passing between components of the same process
- Using an instance of a Local Broadcast Manager



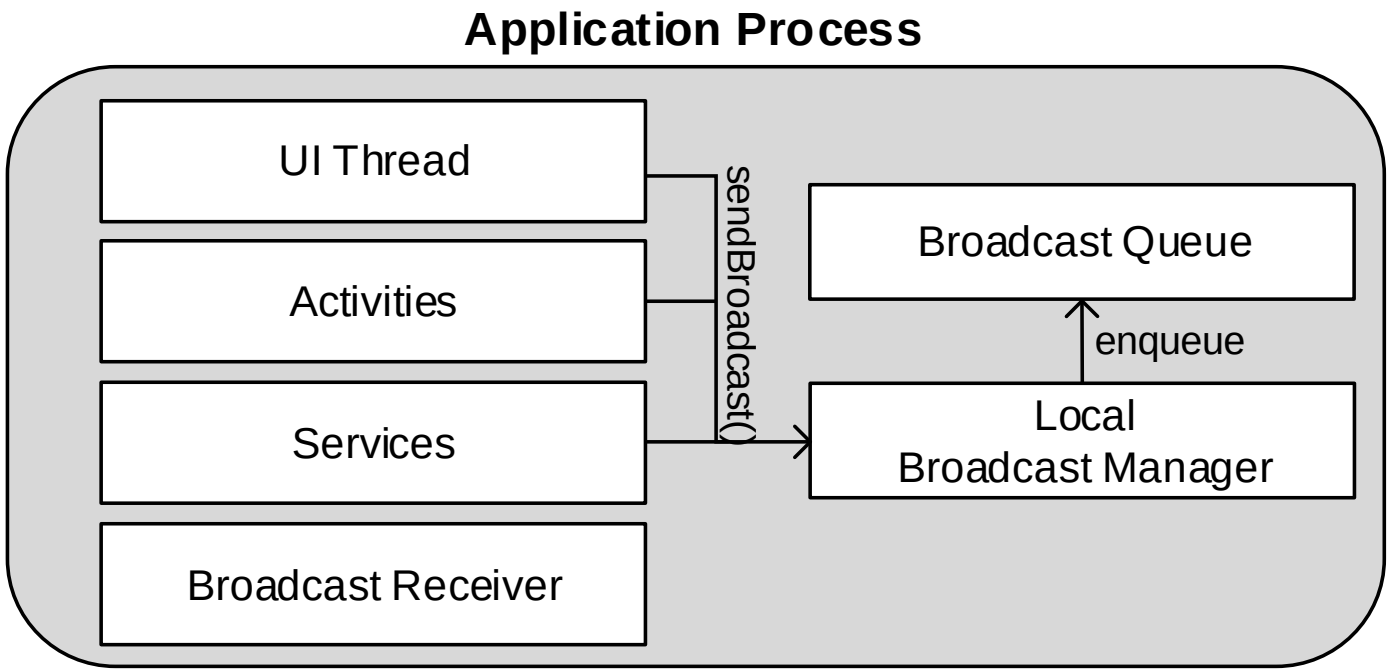
Asynchronous Local Messaging

- Intent passing between components of the same process
- Using an instance of a Local Broadcast Manager
  1. Intent transmission via sendBroadcast()



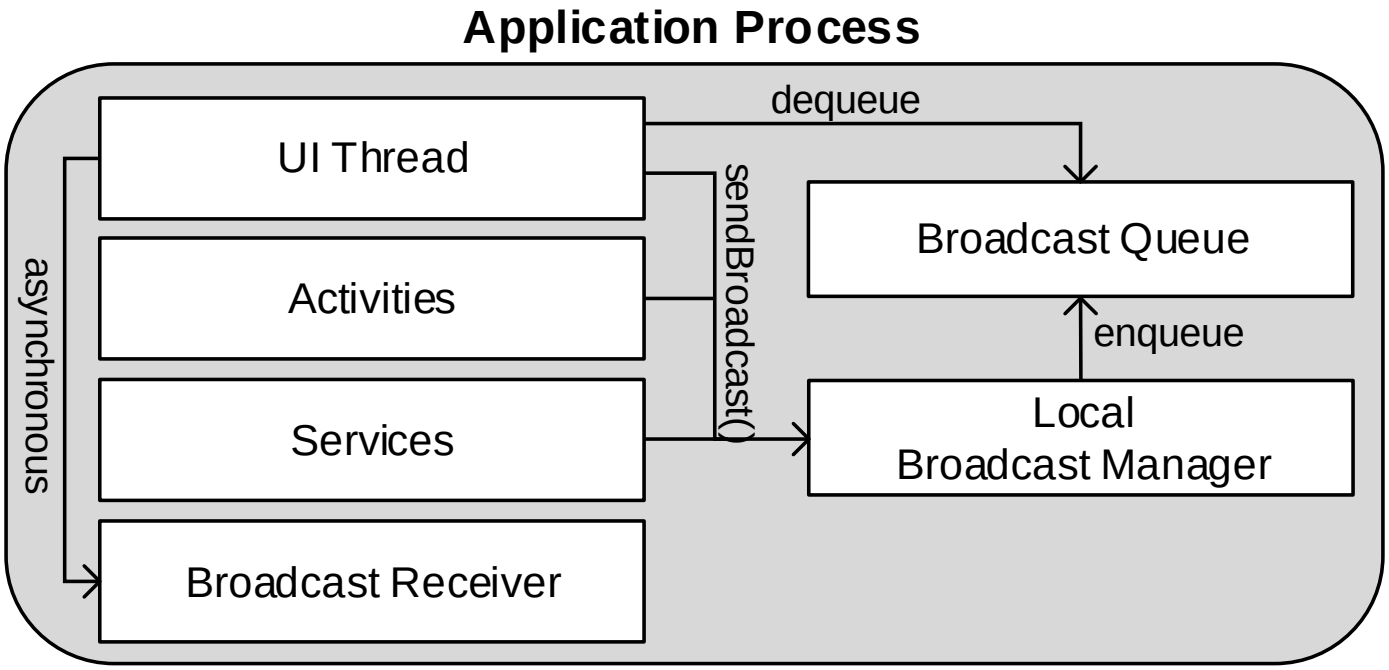
Asynchronous Local Messaging

- Intent passing between components of the same process
- Using an instance of a Local Broadcast Manager
  1. Intent transmission via sendBroadcast()
  2. Objects are stored to a FIFO queue



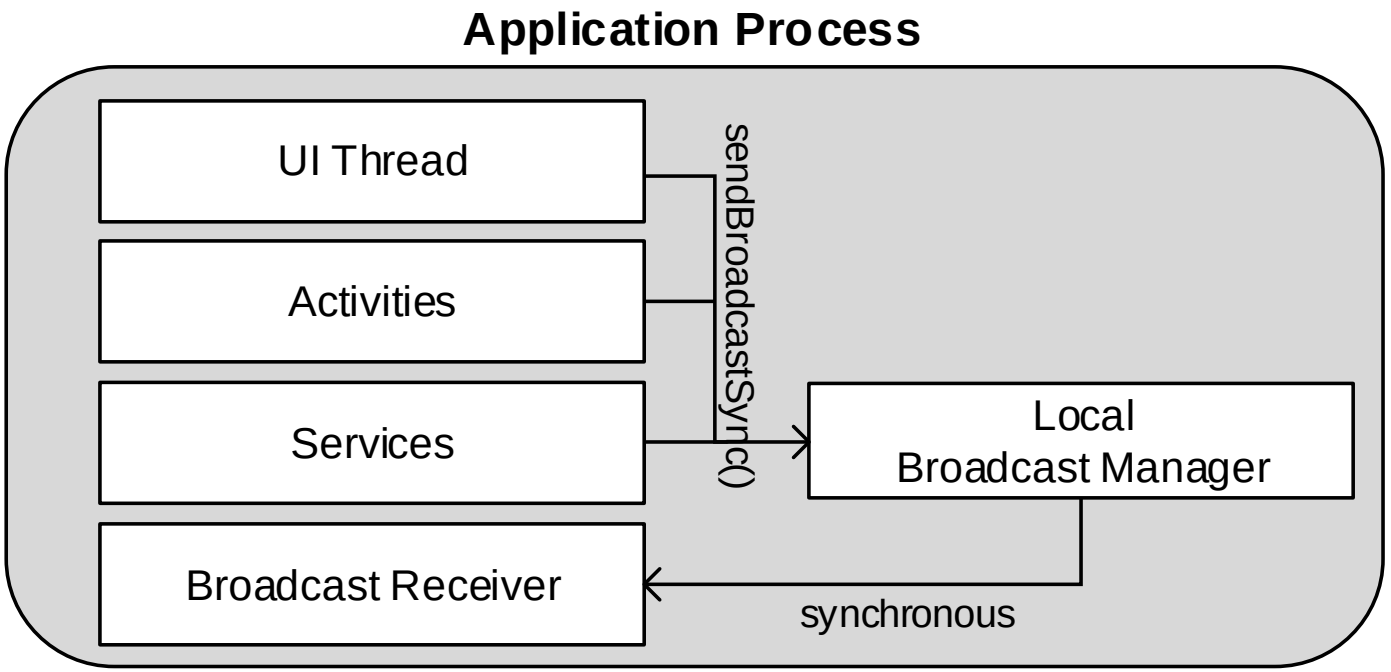
Asynchronous Local Messaging

- Intent passing between components of the same process
- Using an instance of a Local Broadcast Manager
  1. Intent transmission via sendBroadcast()
  2. Objects are stored to a FIFO queue
  3. The delivery is performed by the application's UI thread



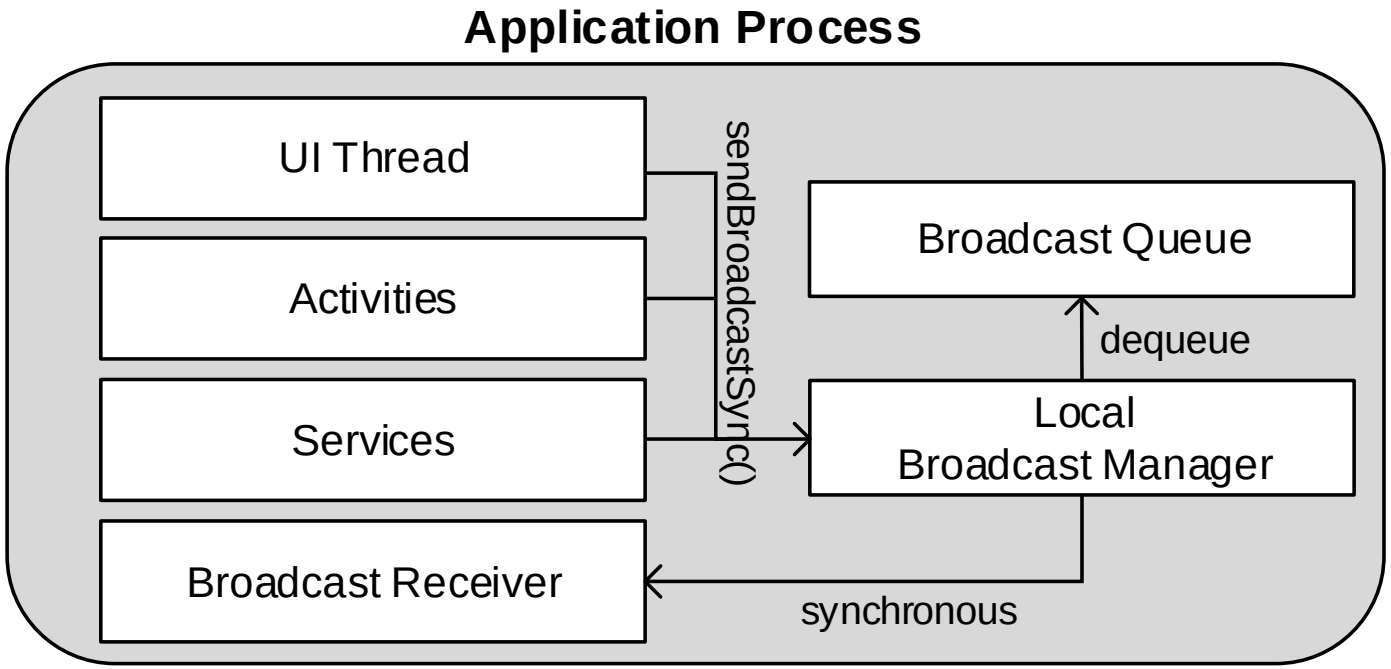
Synchronous Local Messaging

- Intent passing between components of the same process
- Using an instance of a Local Broadcast Manager
  1. Intent transmission via sendBroadcastSync()
  2. Delivery in context of the calling thread



Synchronous Local Messaging

- Intent passing between components of the same process
- Using an instance of a Local Broadcast Manager
  1. Intent transmission via sendBroadcastSync()
  2. Delivery in context of the calling thread
  3. However: all enqueued Intents are processed first

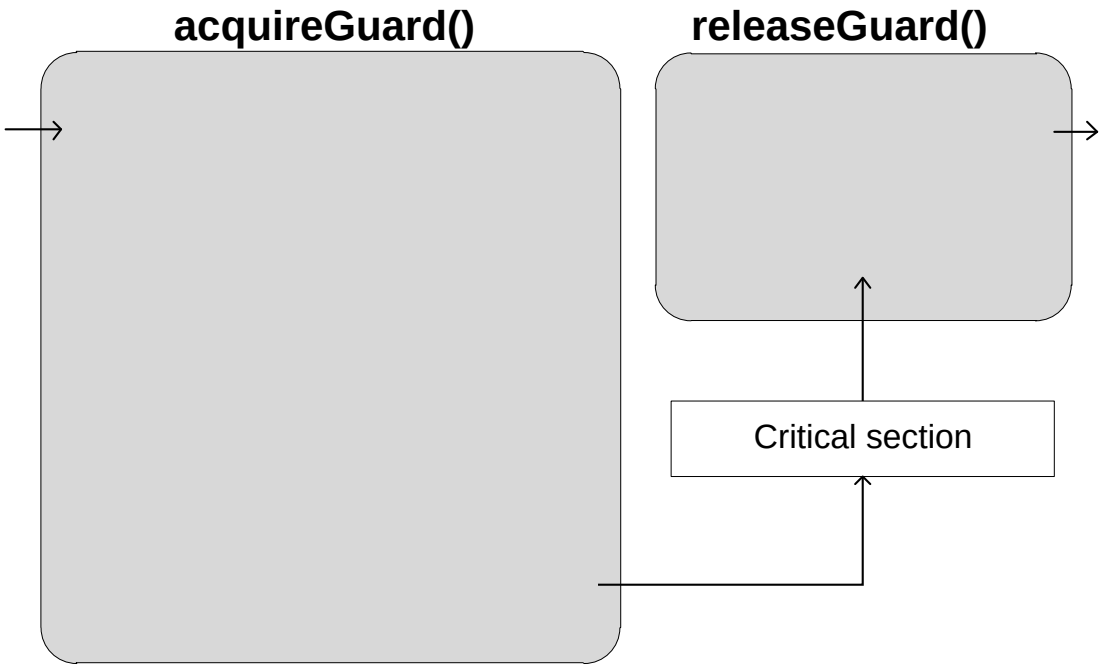


Idealized Approach

- Introduce prioritizing mechanism
  - Inherit Intent priority from sender thread implicitly
  - Replace FIFO storage by sorted queue
- Dedicated real-time threads for Intent delivery
  - Precise polling schedule or notification scheme possible
- Dissolve critical sections for better preemptibility
  - Preprocessing based on local copies instead of global flags
  - Concurrent real-time GC ensures non-disruptive execution

Workaround for Global Broadcasts

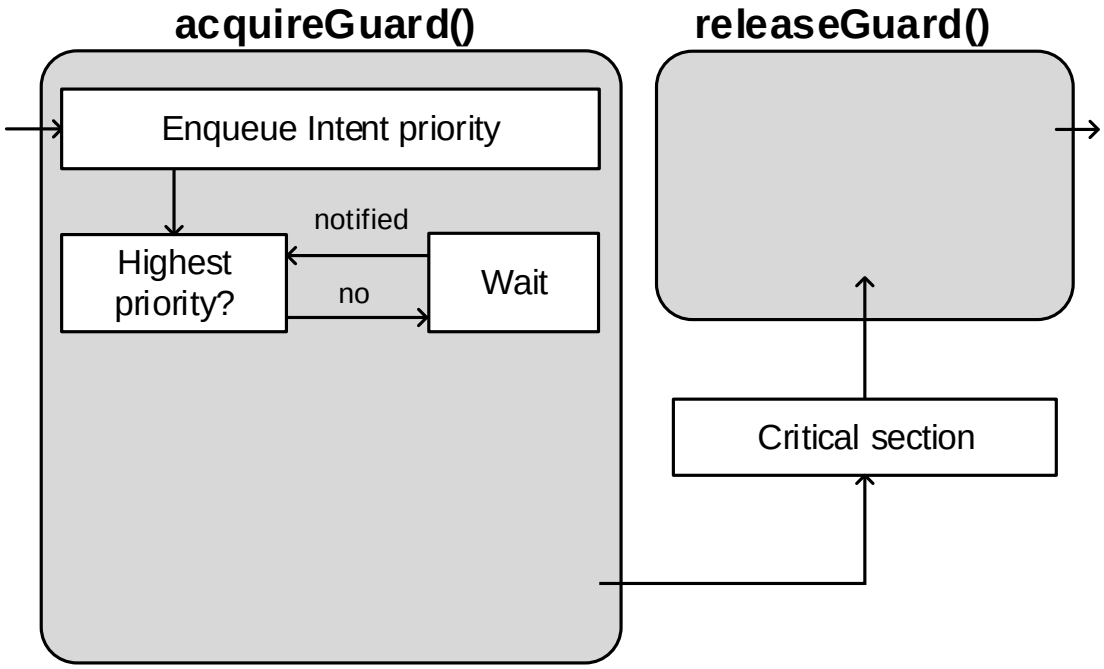
- Complexity of critical sections in AMS is very high
  - Shortening alone seem not sufficient
- Priority-based access management for processing threads



Workaround for Global Broadcasts

- Complexity of critical sections in AMS is very high
  - Shortening alone seem not sufficient

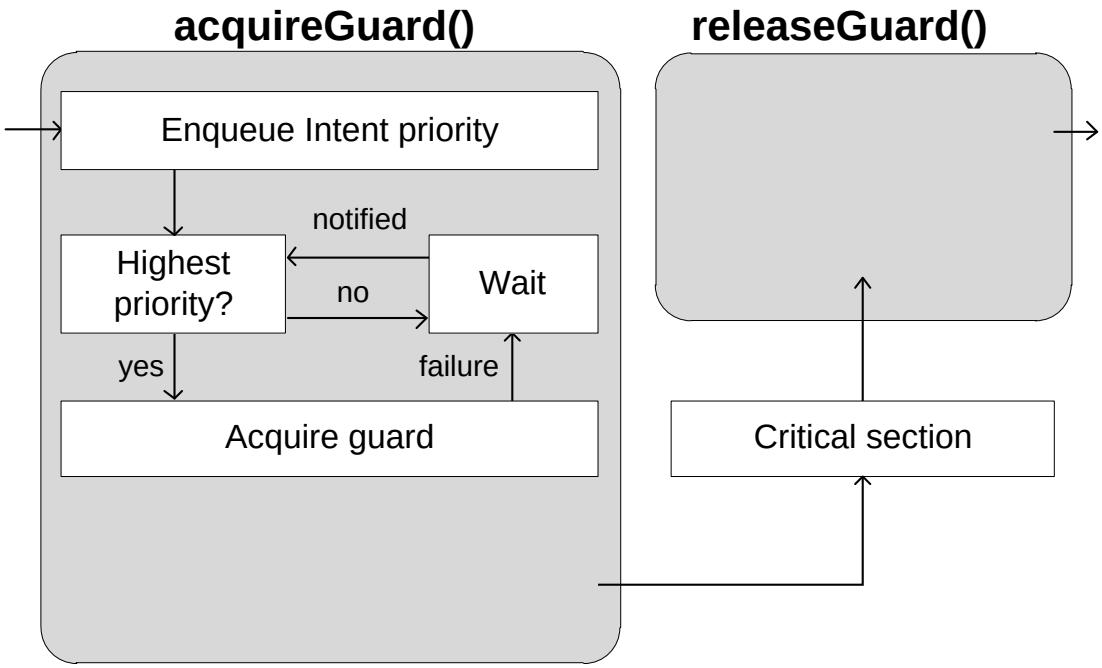
- Priority-based access management for processing threads
  1. Check / enqueue own priority value



Workaround for Global Broadcasts

- Complexity of critical sections in AMS is very high
  - Shortening alone seem not sufficient

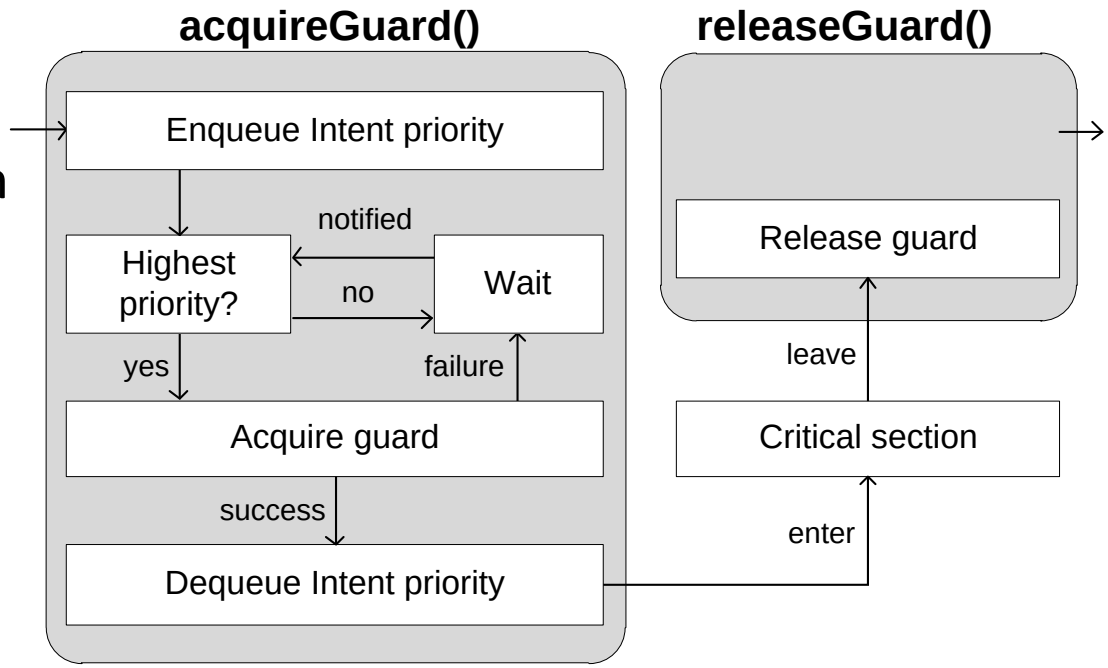
- Priority-based access management for processing threads
  1. Check / enqueue own priority value
  2. Acquire the guard



Workaround for Global Broadcasts

- Complexity of critical sections in AMS is very high
  - Shortening alone seem not sufficient

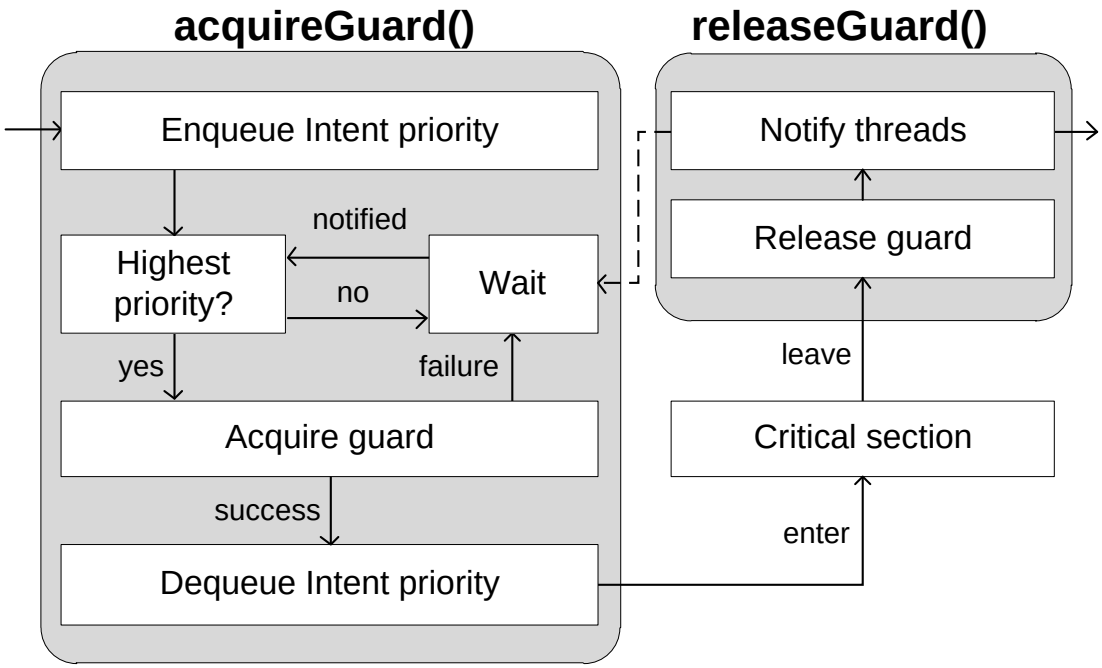
- Priority-based access management for processing threads
  1. Check / enqueue own priority value
  2. Acquire the guard
  3. Dequeue own priority
  4. Run the critical section
  5. Release the guard



Workaround for Global Broadcasts

- Complexity of critical sections in AMS is very high
  - Shortening alone seem not sufficient

- Priority-based access management for processing threads
  1. Check / enqueue own priority value
  2. Acquire the guard
  3. Dequeue own priority
  4. Run the critical section
  5. Release the guard
  6. Notify waiting threads



Modification Overview

- Implementation according to the idealized approach
- Global (parallel) broadcasts via AMS
  - Reduced critical section with ordered access
  - New BroadcastQueue class with priority-based data structure
  - Dedicated real-time thread for broadcast processing
- Local broadcasts via LBM
  - (Almost) removed critical sections by using additional allocations
  - Priority queues instead of FIFO queues
  - Real-time prioritized UI thread for local Intent delivery
  - Synchronous delivery limited to the current Intent object

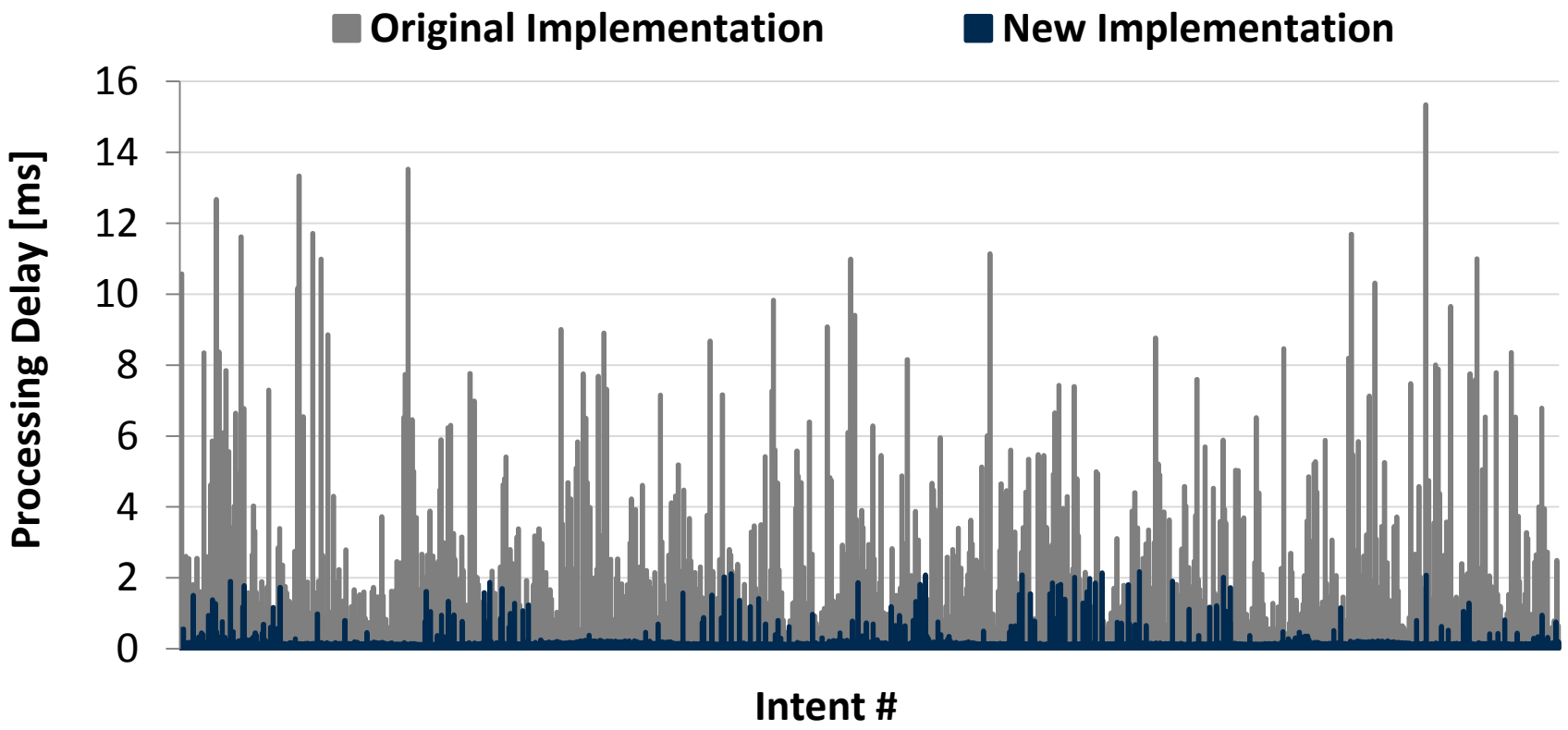
Evaluation – Setup

- Google Nexus 10 with RT extension based on Android 4.2.2
  - Exynos 5259 1.7 GHz dual-core Cortex-A15 CPU & 2 GB RAM

- Testing framework as an Android application
  - Reference-counting GC and RT thread priorities
  - 1 real-time thread & k_{nrt} regular threads
  - N transmitted Intents per thread in total
  - 1 broadcast receiver (does only time measurement)
  - Time measurements for real-time Intents only

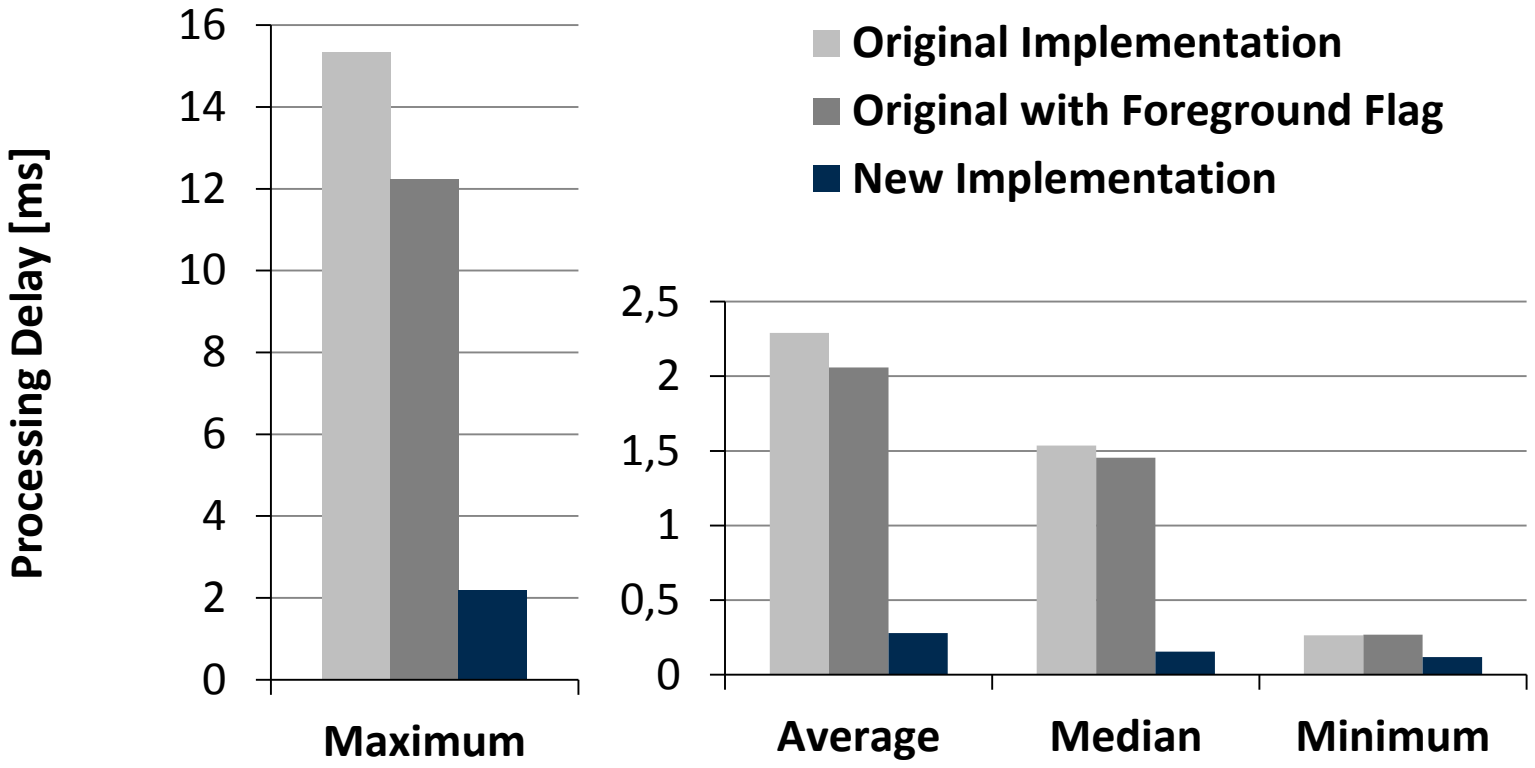
Evaluation – Global Parallel Broadcasts

- Transmission delay in Activity Manager Service
  - $k_{nrt} = 10$
  - $N = 1{,}000$



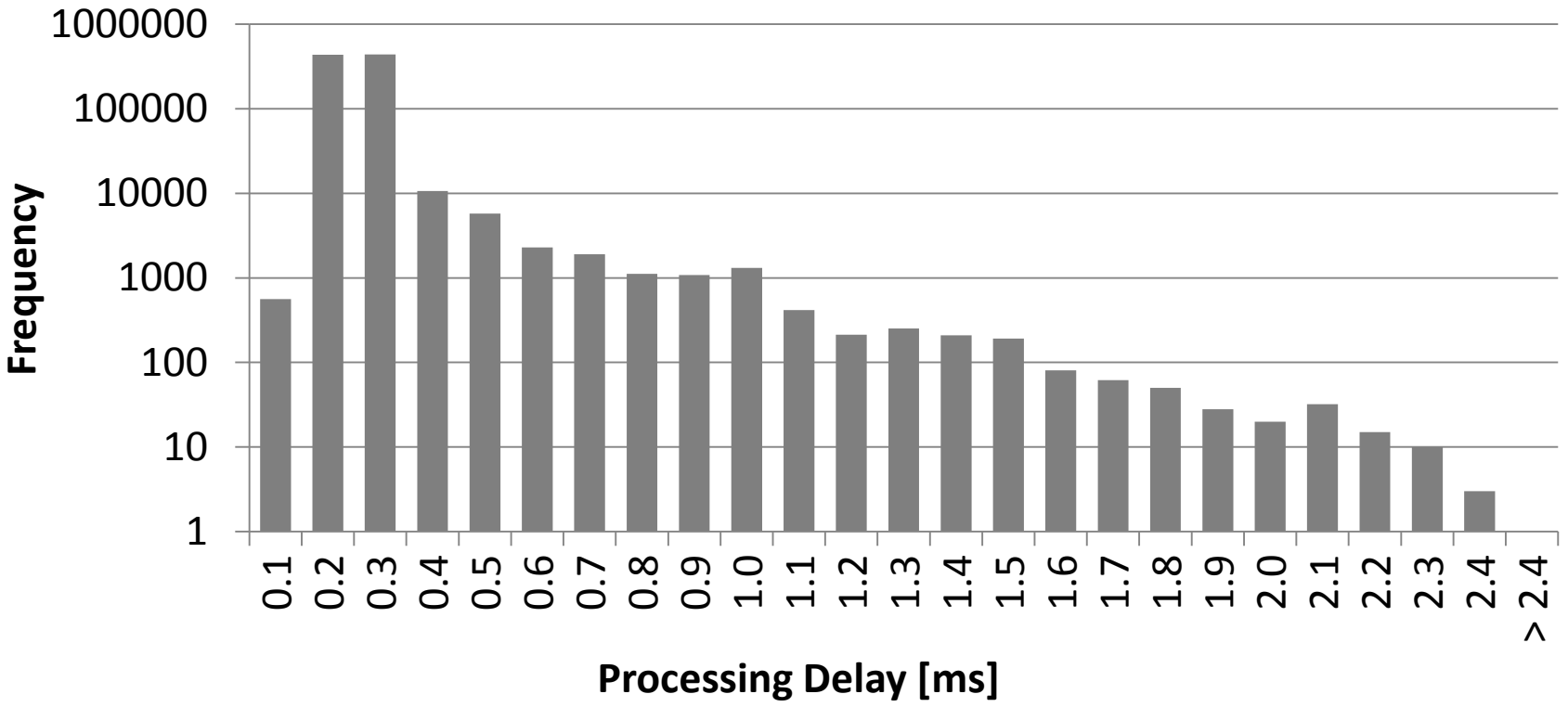
Evaluation – Global Parallel Broadcasts

- Transmission delay in Activity Manager Service
  - $k_{nrt} = 10$
  - $N = 1{,}000$



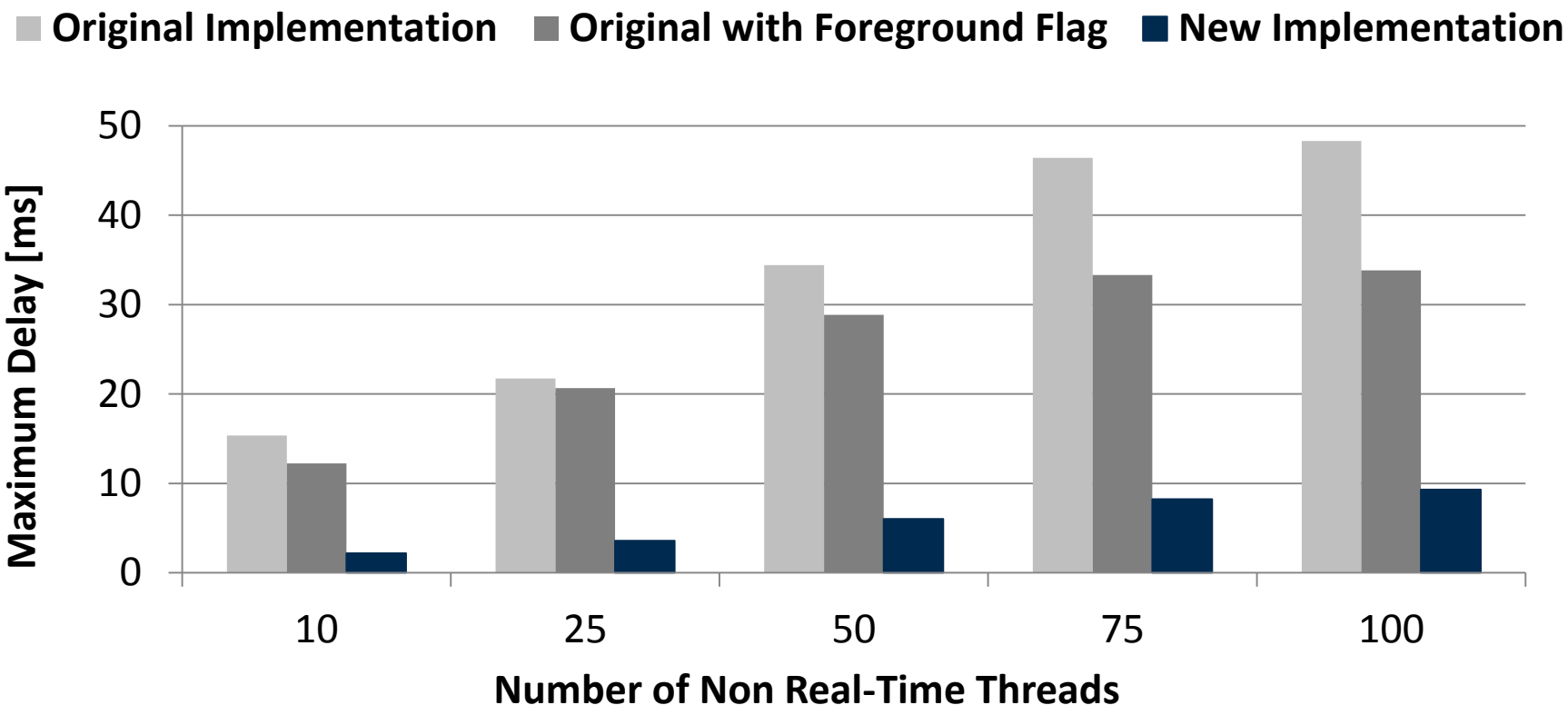
Evaluation – Global Parallel Broadcasts

- Transmission delay: long-term test
  - $k_{nrt} = 10$
  - $N = 1{,}000{,}000$



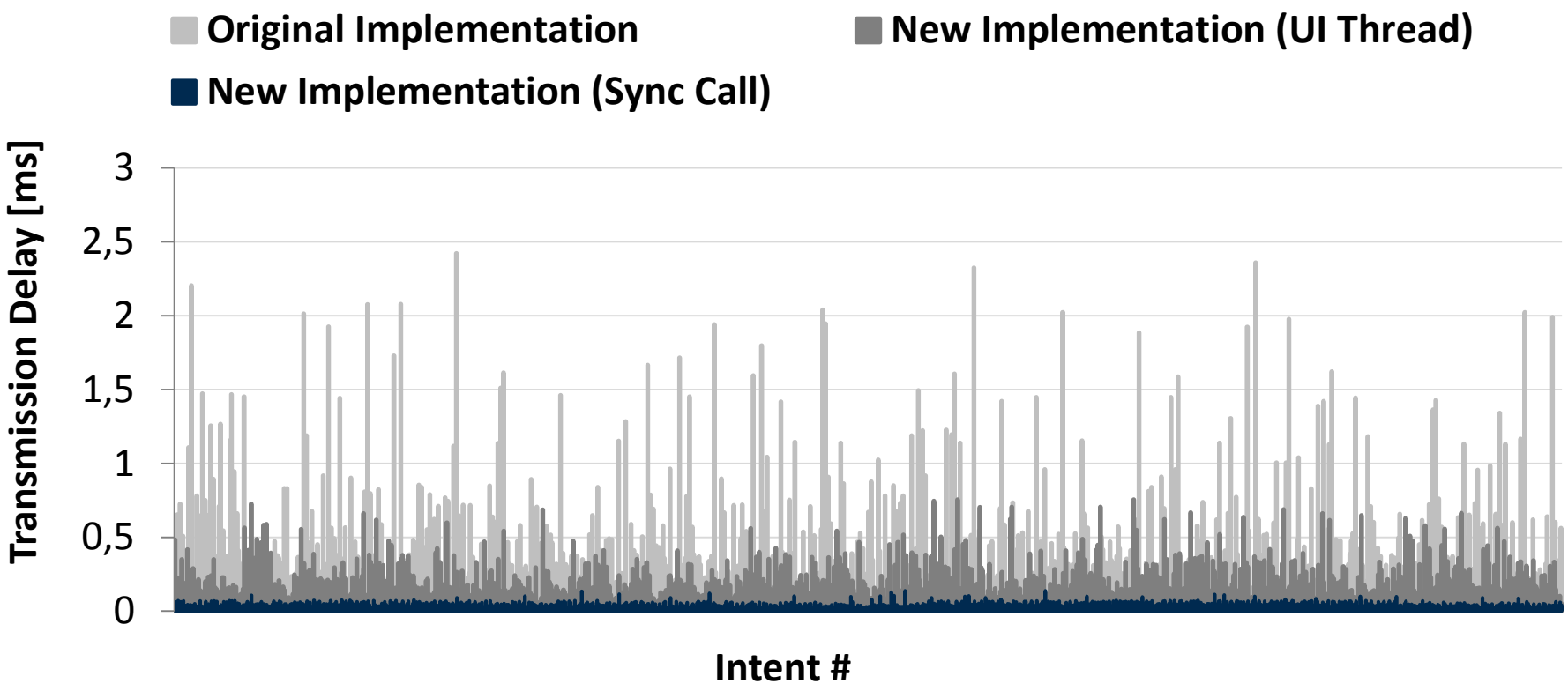
Evaluation – Global Parallel Broadcasts

- Activity Manager Service under load
  - $k_{nrt} \in \{25, 50, 75, 100\}$
  - $N = 1{,}000$



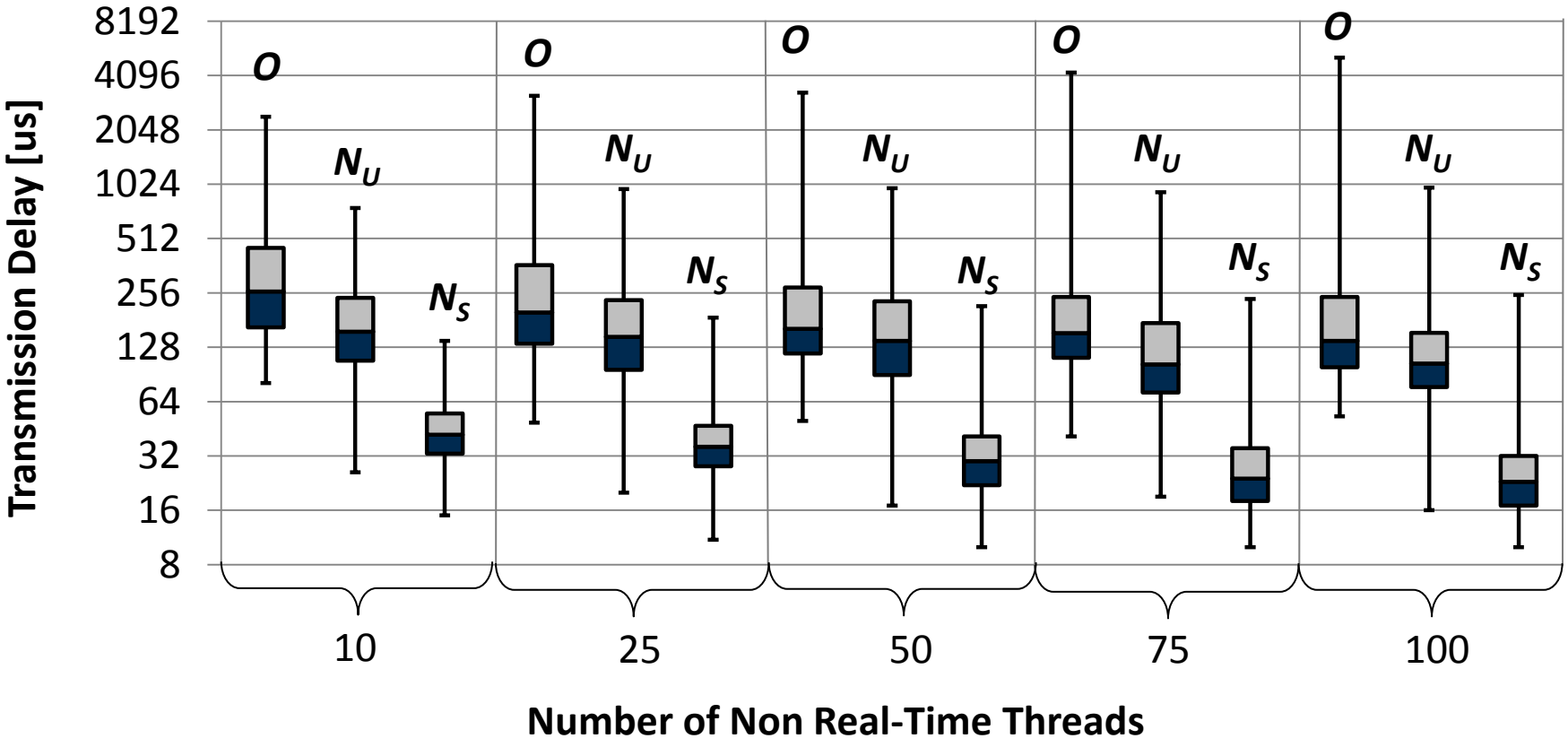
Evaluation – Local Broadcasts

- Transmission delay in Local Broadcast Manager
  - $k_{nrt} = 10$
  - $N = 1{,}000$



Evaluation – Local Broadcasts

- Local Broadcast Manager under load
  - $k_{nrt} \in \{25, 50, 75, 100\}$
  - $N = 1{,}000$



Future Work

- Further elimination of critical sections in the AMS
  - Requires detailed analysis due to Android's high complexity
- Evaluate other Intent types (sticky, ordered)
  - May require additional system or third-party components
  - Relevance for real-time systems not clear
- Using real-time algorithms based on shared memory
  - More appropriate for inter-process communication?

References

[KF12] I. Kalkov, D. Franke, J. F. Schommer, S. Kowalewski. “A Real-time Extension to the Android Platform”. JTRES '12.

[GK13] T. Gerlitz, I. Kalkov, J. Schommer, D. Franke, S. Kowalewski. “Non-Blocking Garbage Collection for Real-Time Android”. JTRES '13.

[YK13] Y. Yan, S. H. Konduri, A. Kulkarni, V. Anand, S. Y. Ko, L. Ziarek. “RTDroid: A Design for Real-Time Android. JTRES '13.

[MH09] W. Mauerer, G. Hillier, J. Sawallisch, S. Hönick, S. Oberthür. “Real-Time Android: Deterministic Ease of Use”. ELCE '12.

[MN10] C. Maia, L. Nogueira, L. M. Pinho. “Evaluating Android OS for Embedded Real-Time Systems”. OSPERT '10.

[K13] K. Kolek. “Application of Android OS as Real-Time Control Platform”. Automatyka / Automatics, volume 17, 2013.

[CF13] E. Chin, A. P. Felt, K. Greenwood, D. Wagner. “Analyzing Inter-application Communication in Android”. MobiSys '11.

[MM10] B. S. Mongia, V. K. Madisetti. “Reliable Real-Time Applications on Android OS”. Whitepaper. 2010.

[PF12] L. Perneel, H. Fayyad-Kazan, M. Timmerman. “Can Android be Used for Real-Time Purposes?”. ICCSII '12.