

# Table of Contents

<b>Summary of talk with Eukeni about future test with Infiniband and Ominpath.....</b>	<b>1</b>
Tests con Omnipath.....	1
Tests con Infiniband.....	1
Observaciones para la simulacion.....	1

# Summary of talk with Eukeni about future test with Infiniband and Ominpath

Tuve una charla muy cortita con Eukeni para ponerme al tanto sobre los tests que van a hacer con Omnipath e Inifiband. Tengo que volver a hablar para ondar en los detalles que necesito para simular, pero igual sirvo bastante para indentificar algunas cosas.

## Tests con Omnipath

En este momento Jorn esta haciendo pruebas con unos nodos que consiguieron con Omnipath. En principio esta probando la librería de comunicación que desarrollo, pero no estan consiguiendo una buena performance. No lograron hacer andar los links mas alla del 50% de la velocidad publicada.

**Conclusion:** Omnipath esta todavia muy verde. La principal ventaja que tendría sobre Inifiband es el precio (que se va a traducir en mayor costo de operación y dolores de cabeza)

## Tests con Infiniband

Dentro de muy poco, posiblemente ppios de Agosto (ya estan todos los permisos y pedidos hechos), van a tener acceso a un cluster Infiniband de la universidad de Albacete (España).

Jorn va a hacer pruebas de la libreria de comunicacion, Tommaso de la topologia, configuracion de la red y flujo de datos.

Las pruebas que van a realizar ya están bastante bien definidas y discutidas con la gente de Albacete. Algunas cosas (como por ejemplo topologia y flujos) se pueden ver en [albacete.pdf](#)

## Observaciones para la simulacion

Un primer buen objetivo sería poder reproducir las pruebas de Infiniband con simulacion. Esto seria para verificar si el nivel de abstracción reproduce el comportamiento que queremos.

Para esto me puse a estudiar un poco Infiniband e identifiqué algunos puntos débiles de la simulacion actual.

- **Ruteo:** Infiniband provee por default 9 algoritmos de ruteo. Uno de los tests previstos es si estos proveen un buen balance de carga entre los links redundantes.  
En la simulacion ahora podemos definir cualquier cantidad de flujos y rutas arbitrarias. Pero se define una por una "manualmente" en el codigo. O sea que ahora si conocemos empiricamente que rutas se definen lo podemos replicar en la simulacion, pero no podemos predecir de antemano las rutas (mucho menos balance de carga, etc).
- **QoS (prioridad a flujos):** una de las pruebas previstas y algo importante para PhaseI es estudiar como garantizar la QoS para cada uno de los flujos de PhaseI.  
En la simulacion todavia no hicimos nada sobre esto (con MVA en ppio no se puede).  
En las notas hay un whitepaper que cuenta bien como trabaja el QoS de infiniband. Suena que es un muy buen punto para estudiar con simulacion.
- **Congestion Control (CC):** Infiniband implementa control de congestión marcando los paquetes en los switches y notificando a la fuente (closed loop feedback control system). No debería haber descartes (loss-less) salvo por fallas de hardware. También implica que los switches Infiniband tiene buffers chiquitos porque el mayor encolamiento se hace en la fuente (en las PCs). Se implementa por flujo (relacionado con QoS).  
En la simulacion no tenemos nada de CC, asumimos buffers infinitos (seria para ver cuanto buffer es necesario). Obviamente esta política de encolamiento en el source (en vez de los switches) impacta la latencia, asi que seria importante creo tenerla en la simulacion.  
El whitepaper de las notas explica bien el CC de Infiniband y hay un paper de simulacion con OMNET++ sobre CC.

Un punto importante a tener en cuenta es que ninguna de estas cosas es facil de representar con MVA.

Aqui hay algunas notas sobre Infiniband

-- MatiasAlejandroBonaventura - 2016-07-07

---

This topic: Main > PhaseIMeetingWithEukeni

Topic revision: r1 - 2016-07-07 - MatiasAlejandroBonaventura



Copyright &© 2008-2020 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback