Virtual Debugging Network for Embedded Systems

Name: Vikas Varshney

Title: Senior Software Engineer
Company Name: Texas Instruments
Email: varshney@ti.com

SPRP510

Agenda

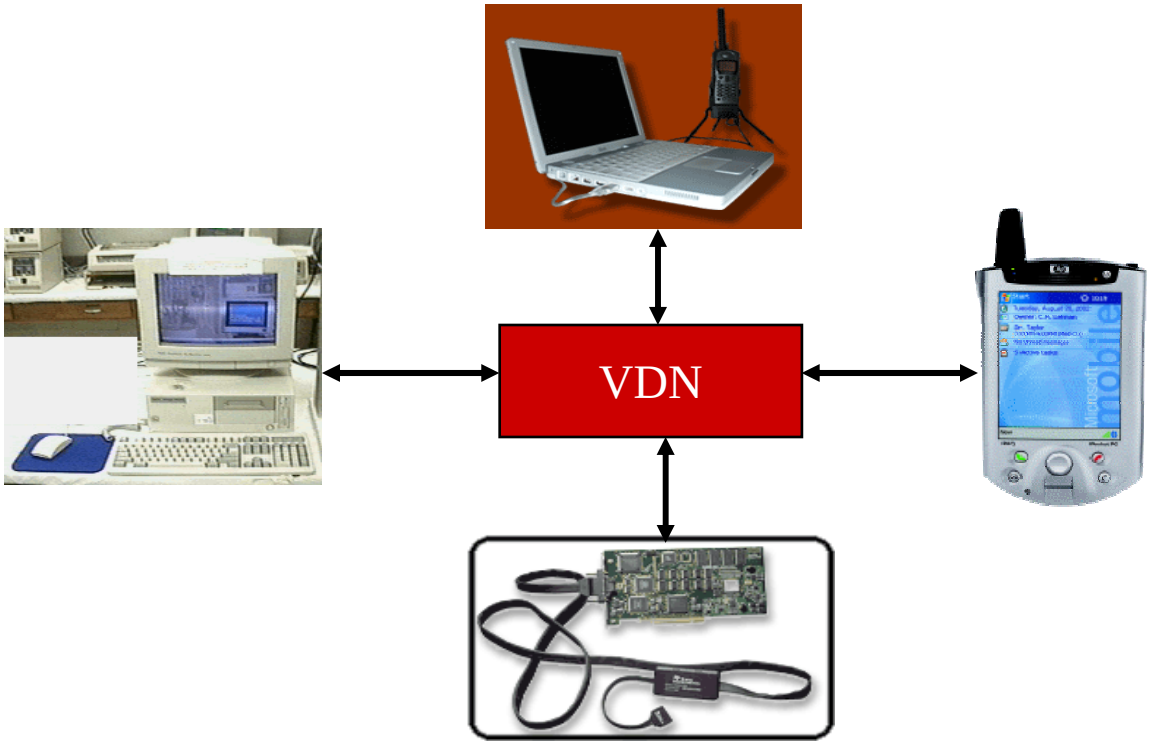
- **Introduction**
  - Overview
- **Technology**
  - Design Principle
  - TI Tooling Support
- **Advantages**
  - Distributed Development
  - Heterogeneous Tooling
  - Shared Resources

Introduction - Overview

- Virtual multi-client, multi-target debugging
- Seamless local/remote debugging
- Heterogeneous platform tooling
- No heavyweight virtualization software
- Optimized Remote Procedure Calls
- Built-in with TI Target Server Technology

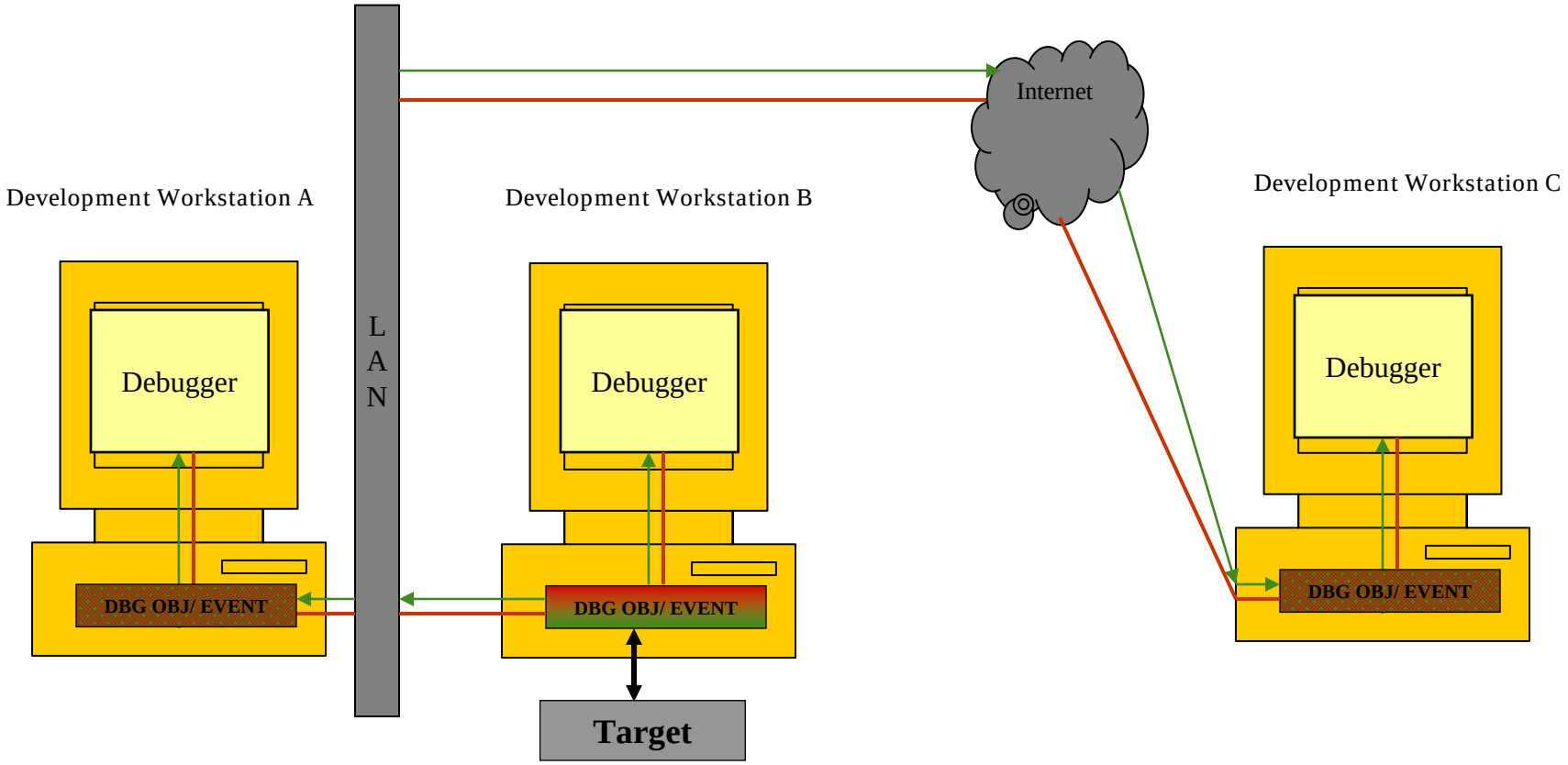
Introduction - Overview

- **Virtual debug network (VDN) components**
  - VDN enabled debug tools
  - Targets
  - Network connection



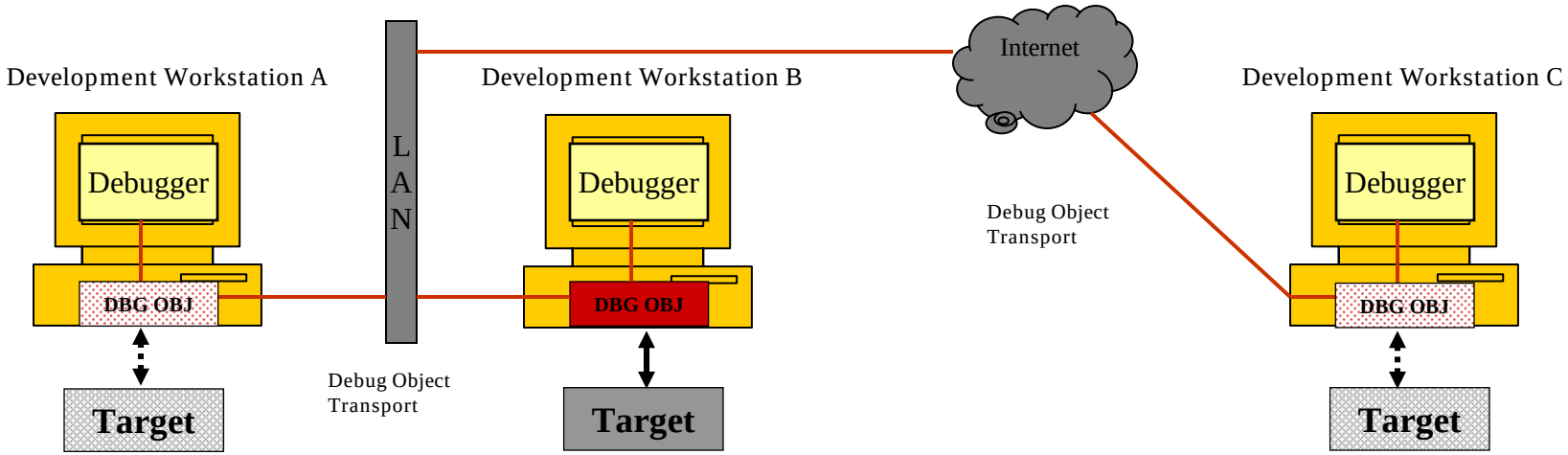
Technology - Design

- Distributed debug object
- Debug events



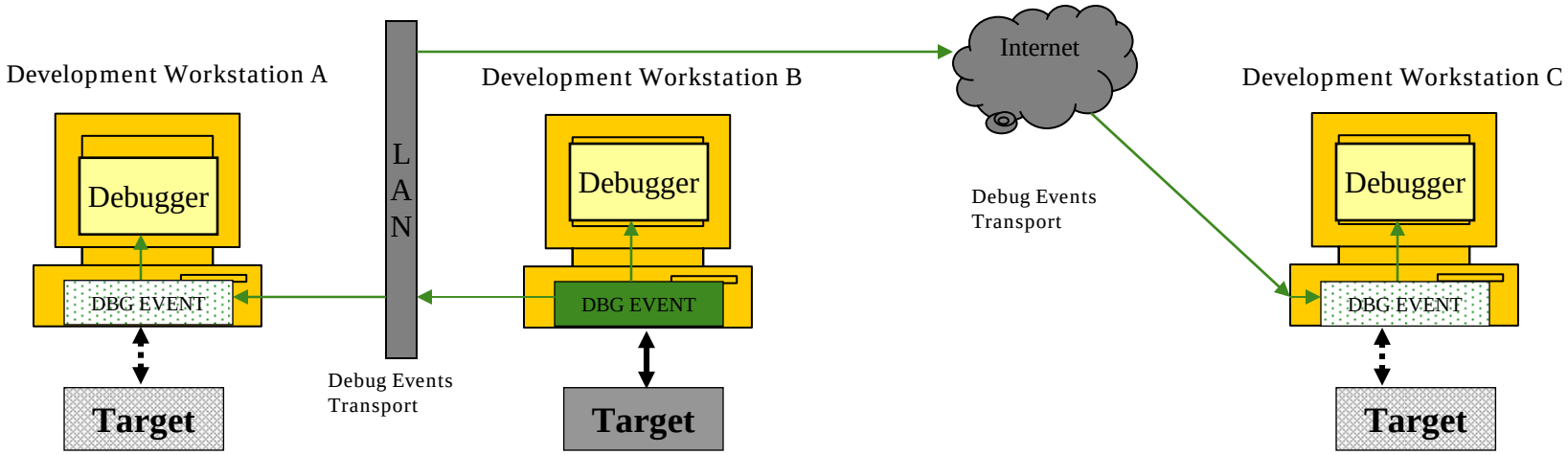
Technology - Design

- **Distributed debug object**
  - Encapsulates debug features
    - read/write registers, memory, execution control etc
  - Exports C/C++ APIs for debug features
  - API published for remote access
  - Automatic discovery
  - Request serialization and synchronization
  - Exclusive access control



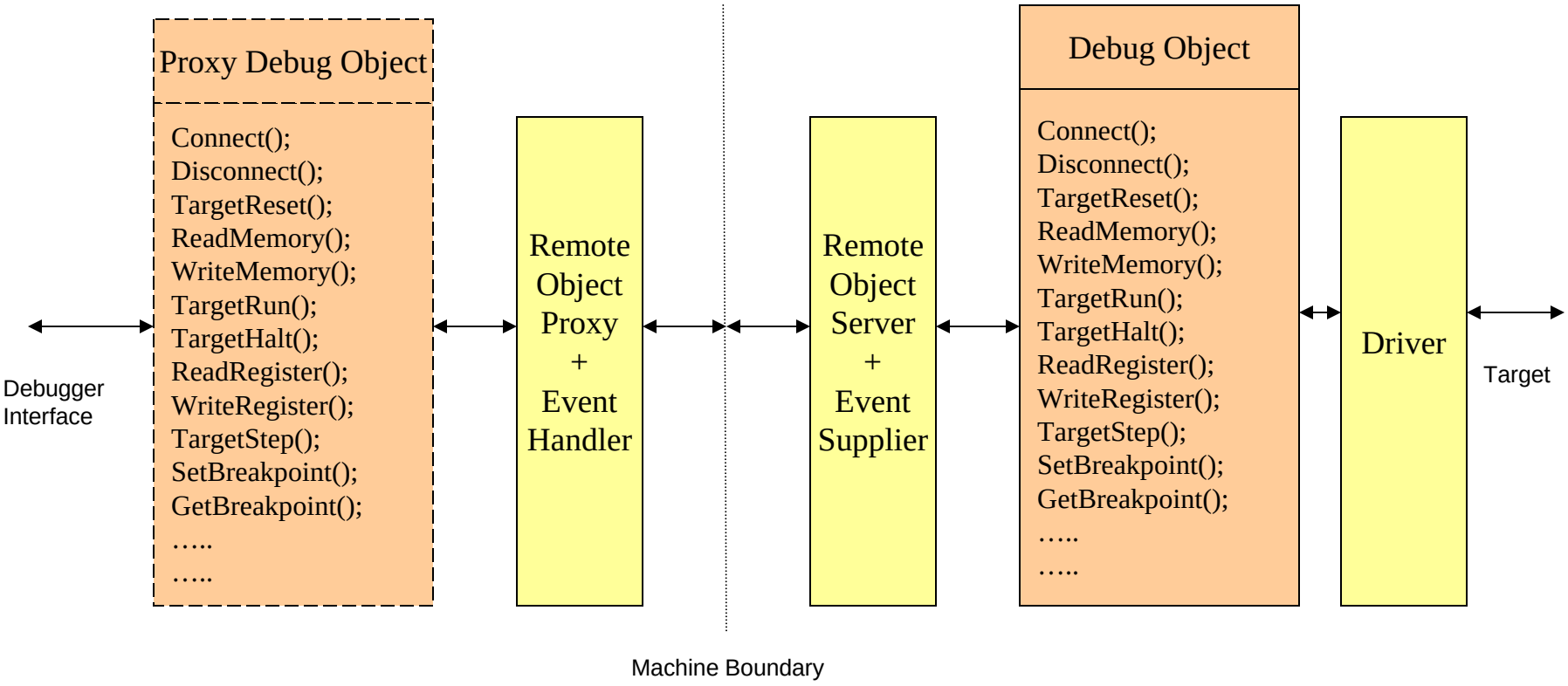
Technology - Design

- **Debug events**
  - Supplier-consumer model
  - Asynchronous target state and operation events
    - Target run, halt, register change, breakpoint hit etc.
  - Globally available target events
  - Event supplier - debug object
  - Event consumer – debugger, tools, client



Technology - Design

- Server-proxy model for debug object and events



Technology – TI Tooling Support

- **TI Target Server v3.0**
  - Emulation/Simulation interface
  - 3rd party debugger integration
- **Seamless APIs for local or remote target**
  - Capability to connect to remote target
  - Capability to attach to an existing target instance
- **Support**
  - Hosts
    - Windows and Linux
  - Targets
    - C6xx, C55x, C54x, C2x, ARM9, AMM11 and MSP430
- **Code Composer Essentials (CCE) v1.0, v2.0**

Advantages – Distributed Development

- **Globally distributed team collaboration**
  - Development teams can virtually work on same resources.
  - Developers can share debug sessions across sites.
- **Efficient remote debugging**
  - Users can debug remotely as they are debugging locally.
  - No additional software or virtualization tools required.
- **Remote diagnosis**
  - No need for physically transfer hardware for debug across sites.
- **Time effectiveness**
  - Multiple users can effectively timeshare debug hardware.
  - Difficult to reproduce bugs can easily be reproduced, shared and fixed in a remote environment.

Advantages – Heterogeneous Tooling

- **Tools reusability across platforms**
  - Windows only debug tools can be used for Unix drivers.
- **Network access to non-network emulators**
  - PCI or USB emulators can be accessed via network without any inbuilt network support.
- **Debuggers interoperability**
  - Debugger ‘A’ can work with debugger ‘B’ for complex or compliment debugging.
  - Debuggers are aware of target states being changed or alerted by another debugger making them synchronized with each other

Advantages – Shared Resources

- **Shared debug target hardware**
  - Fewer debug targets can be shared among multiple users by setting up a ‘target farm’.
- **Distributed test infrastructure**
  - Tests can be run on a shared target remotely.
  - Tests can be run in parallel using multiple shared targets.
- **Cost effectiveness**
  - Fewer hardware resources.
  - Reduced debug and diagnosis time.

TI Developer Conference

Virtual Debugging Network for Embedded Systems

Presenter Name: Vikas Varshney

Title: Senior Software Engineer
Company Name: Texas Instruments
Email Address: varshney@ti.com

SEE THE FUTURE

CREATE YOUR OWN