SQL Server Internals: The Practical Angle

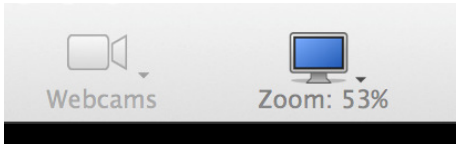
Sneak Peek

Dmitri Korotkevitch
Moderated by Roberto Fonseca

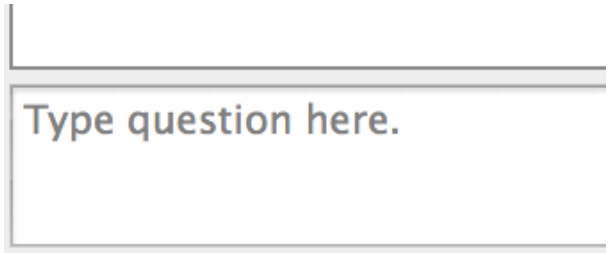
PREVIEW EDITION

Microsoft idera DELL Software

Technical Assistance



Maximize your screen with the zoom button on the top of the presentation window



Type your questions in the question pane on the right side

Thank You

Presenting Sponsors

Gain insights through familiar tools while balancing monitoring and managing user created content across structured and unstructured sources.

www.microsoft.com

Idera is a leading provider of IT performance monitoring solutions.

www.idera.com

Solutions from Dell help you monitor, manage, protect and improve your SQL Server environment.

www.software.dell.com

Supporting Sponsors

October 27-30 / Seattle, WA

Planning on attending PASS Summit 2015? Start saving today!

- The world's largest gathering of SQL Server & BI professionals
- Take your SQL Server skills to the next level by learning from the world's top SQL Server experts, in over 190 technical sessions
- Over 5000 registrations from 52 countries

Save $200 right now using discount code **24HOP15**

$2,195

until September 20, 2015

PASS

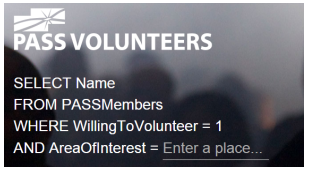
Explore Everything PASS Has to Offer

24 HOURS OF PASS

FREE ONLINE WEBINAR EVENTS

PASS SQL saturday

FREE 1-DAY LOCAL TRAINING EVENTS



VOLUNTEERING OPPORTUNITIES

PASS CHAPTERS

LOCAL USER GROUPS AROUND THE WORLD

PASS VIRTUAL CHAPTERS

ONLINE SPECIAL INTEREST USER GROUPS

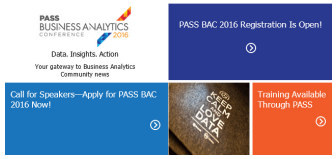
PASS CommunityCONNECTOR

PASS COMMUNITY NEWSLETTER

FREE ONLINE RESOURCES

PASS BUSINESS ANALYTICS 2016

BUSINESS ANALYTICS TRAINING



BA INSIGHTS NEWSLETTER

PASS

Dmitri Korotkevitch Bio

Dmitri is a Microsoft SQL Server MVP and Microsoft Certified Master, author of “Pro SQL Server Internals” and “Expert SQL Server In-Memory OLTP” with over 15 years of experience working with SQL Server as an Application and Database Developer, DBA, and Database Architect. Dmitri specializes in the design, development, and performance tuning of complex OLTP systems that handle thousands of transactions per second around the clock.

@aboutsqlserver

linkedin.com/pub/
dmitri-korotkevitch/5/980/b7

aboutsqlserver.com

SQL Server Internals: The Practical Angle

Sneak Peek

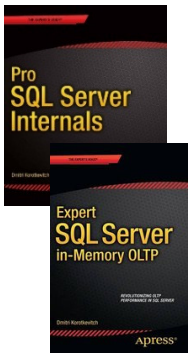
Dmitri Korotkevitch

PREVIEW EDITION

Microsoft idera DELL Software

About me

20+ years of experience in IT
15+ years of experience working with SQL Server
Microsoft SQL Server MVP
Microsoft Certified Master
Author of “Pro SQL Server Internals” and “Expert SQL Server In-Memory OLTP”



Blog: http://aboutsqlserver.com
Email: dk@aboutsqlserver.com

Separation of Duties (Dev vs. DBA)

Pre-Con Goals

Reduce separation of duties in the industry

Provide an overview of how SQL Server components work under the hood and how their behavior affects your systems

Bonus: Teach how to answer “It Depends” to every SQL Server-related question ☺

Agenda

Tables and indexes

- Heap Tables

Internal implementation of database objects

- Triggers and fragmentation

Locking, blocking and concurrency

- Lock escalation

Query execution and plan caching

- Parameter sniffing

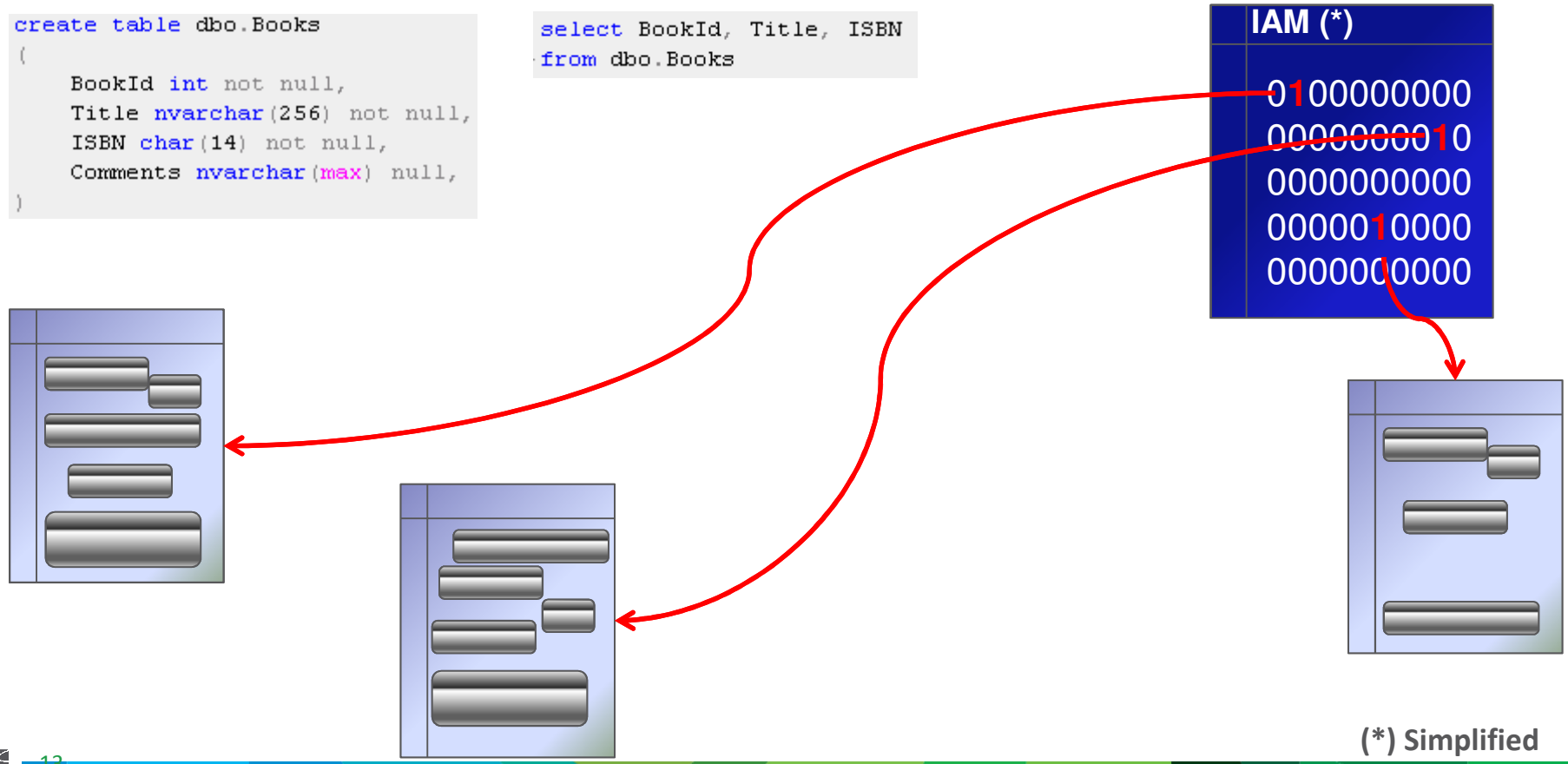
System troubleshooting

Heap Tables

Heap Tables

```
create table dbo.Books
(
    BookId int not null,
    Title nvarchar(256) not null,
    ISBN char(14) not null,
    Comments nvarchar(max) null,
)
```

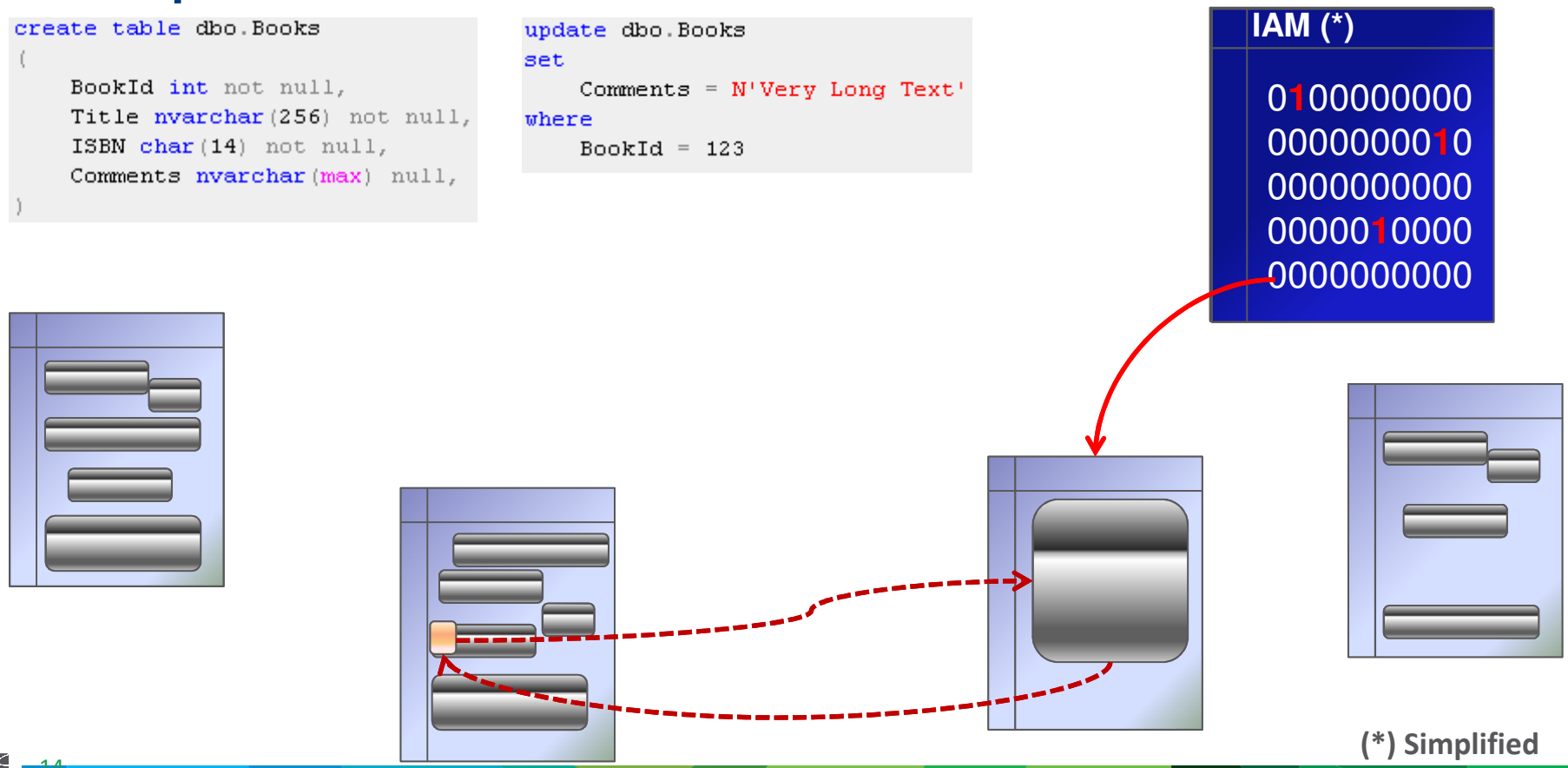
```
select BookId, Title, ISBN
from dbo.Books
```



Heap Tables

```
create table dbo.Books
(
    BookId int not null,
    Title nvarchar(256) not null,
    ISBN char(14) not null,
    Comments nvarchar(max) null,
)
```

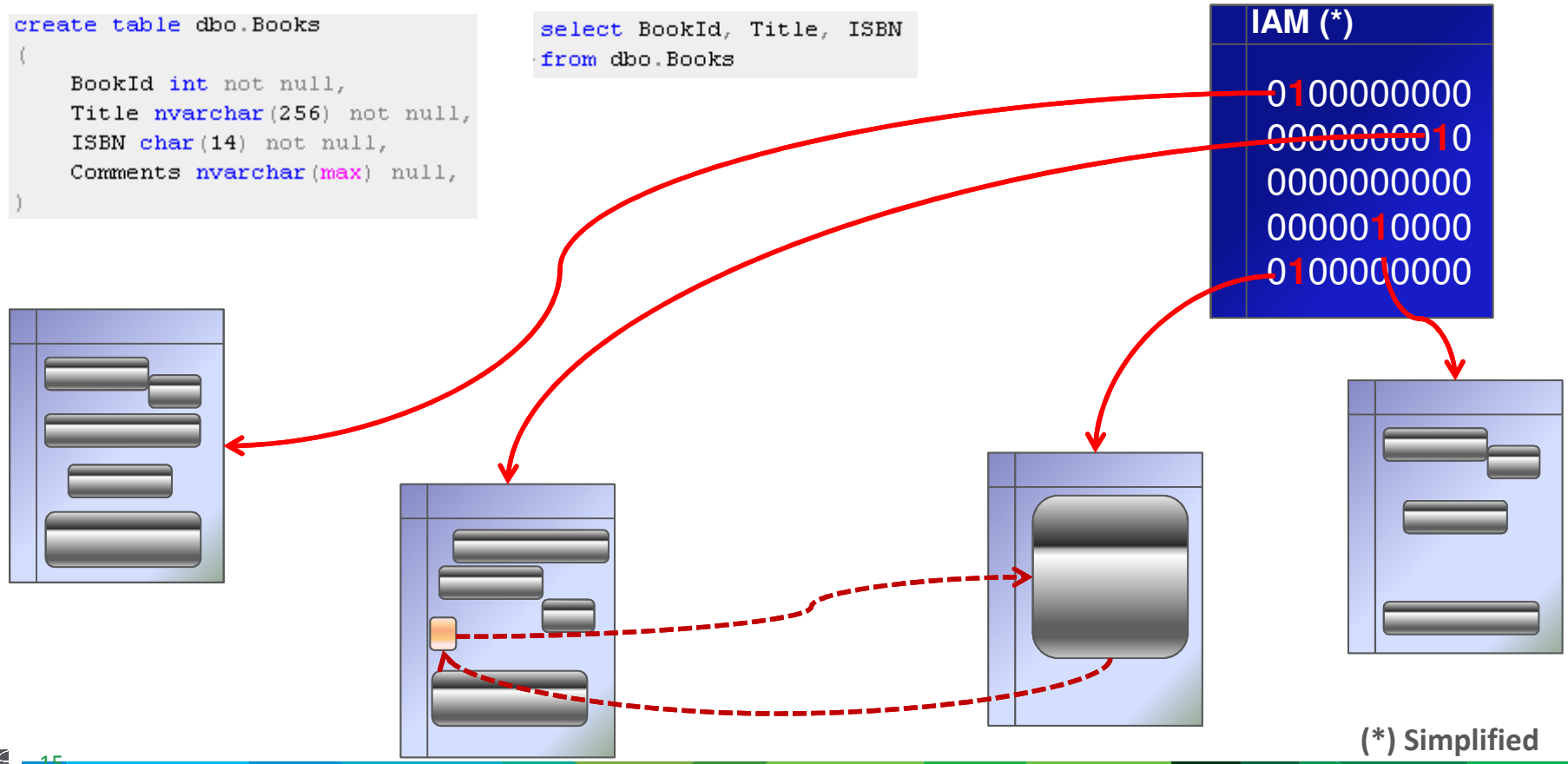
```
update dbo.Books
set
    Comments = N'Very Long Text'
where
    BookId = 123
```



Heap Tables

```
create table dbo.Books
(
    BookId int not null,
    Title nvarchar(256) not null,
    ISBN char(14) not null,
    Comments nvarchar(max) null,
)
```

```
select BookId, Title, ISBN
from dbo.Books
```



Heap Tables and Forwarding Pointers

Demo

Heap Tables

Potential issues

- Forwarding pointers introduce extra I/O
- Suboptimal control of free space
  - Uses PFS to estimate amount of free space on the page

Use-cases

- Staging environments that require fast data load

Triggers

DML Triggers

Triggers degrade performance

ALTER UPDATE/DELETE triggers add to index fragmentation

Avoid time consuming operations and external access

- Transactions are active and all locks are held during trigger execution
- Exceptions can roll back the transaction

CONTEXT_INFO allows to pass parameter(s) to the triggers

Triggers and Index Fragmentation

Demo

Lock Escalation

Exclusive (X) Locks

Acquires by *writers* when row needs to be modified

Always held until the end of transaction regardless of the isolation level

```
set transaction isolation level read committed
begin tran
    update dbo.Orders set Approved = 1 where OrderId = 1001

    select * from sys.dm_tran_locks where request_session_id = @@SPID
```

	resource_type	resource_subtype	resource_database_id	resource_description	resource_associated_entity_id	resource_lock_partition	request_mode	request_t
1	PAGE		2	1:1438	7566282316938805248	0	IX	LOCK
2	OBJECT		2		218295668	0	IX	LOCK
3	KEY		2	(7d00a258ee47)	7566282316938805248	0	X	LOCK

Shared (S) Locks

Acquired by *readers* when row needs to be read

```
set transaction isolation level repeatable read
begin tran
    select OrderId, OrderNum dbo.
    from dbo.Orders
    where OrderId = 535

    select *
    from sys.dm_tran_locks
    where request_session_id = @@SPID
```

	resource_type	resource_subtype	resource_database_id	resource_description	resource_associated_entity_id	resource_lock_partition	request_mode	request_ty
1	PAGE		2	1:1438	7566282316938805248	0	IS	LOCK
2	OBJECT		2		218295668	0	IS	LOCK
3	KEY		2	(7d00a258ee47)	7566282316938805248	0	S	LOCK

Intent (I*) Locks

Indicate locks on the child objects (page, row)

```
set transaction isolation level read committed
begin tran
    update dbo.Orders set Approved = 1 where OrderId = 1001

    select * from sys.dm_tran_locks where request_session_id = @@SPID
```

	resource_type	resource_subtype	resource_database_id	resource_description	resource_associated_entity_id	resource_lock_partition	request_mode	request_ty
1	PAGE		2	1:1438	7566282316938805248	0	IX	LOCK
2	OBJECT		2		218295668	0	IX	LOCK
3	KEY		2	(7d00a258ee47)	7566282316938805248	0	X	LOCK

Lock Escalation

SQL Server escalates locks to the table/partition level

- After ~5,000 locks per statement per object
- If failed – after ~1,250 new locks per statement per object

It is completely normal unless it is not.. ☺

Pattern: batch operation triggers lock escalation. All other sessions that are trying to obtain incompatible intent lock on the object are blocked.

Lock Escalation

Demo

Lock Escalation

Troubleshooting

- High percent of intent lock waits in the system (LCK_M_I*)
- *Lock Escalation* events in xEvents and SQL Trace
- *Table Lock Escalations/Sec* performance counter

Disabling Lock Escalation

- SQL Server 2008+: *ALTER TABLE .. SET LOCK_ESCALATION*
- TF 1211 / 1224

Alternatively, consider switching to optimistic concurrency

Parameter Sniffing

Plan Caching

SQL Server caches and reuses execution plans based on

- Same batch text (including whitespaces, comments, etc.)
- Same session settings (ANSI NULL, etc.)
- Objects are referenced with their schemas and/or users have the same default schema

Plan caching reduces CPU load; however, it introduces some potential issues

- Possible plan cache pollution and extensive memory usage
- Parameter sniffing
- Possibly sub-efficient plans for some parameter values

Plan Caching Issues

Demo

We covered

Heap tables and forwarding pointer

DML triggers

Lock escalation

Parameter sniffing

Slides and demos are available at: http://aboutsqlserver.com

Questions?

Follow @pass24hop

Share your thoughts with **#pass24hop** & **#sqlpass**

Thank You for Attending

PREVIEW EDITION

Microsoft idera DELL Software

Coming Up Next …

Stress Inoculation:

Maintaining Performance Under Pressure

Russ Thomas

PREVIEW EDITION

Microsoft idera DELL Software

PREVIEW EDITION

PASS SUMMIT