

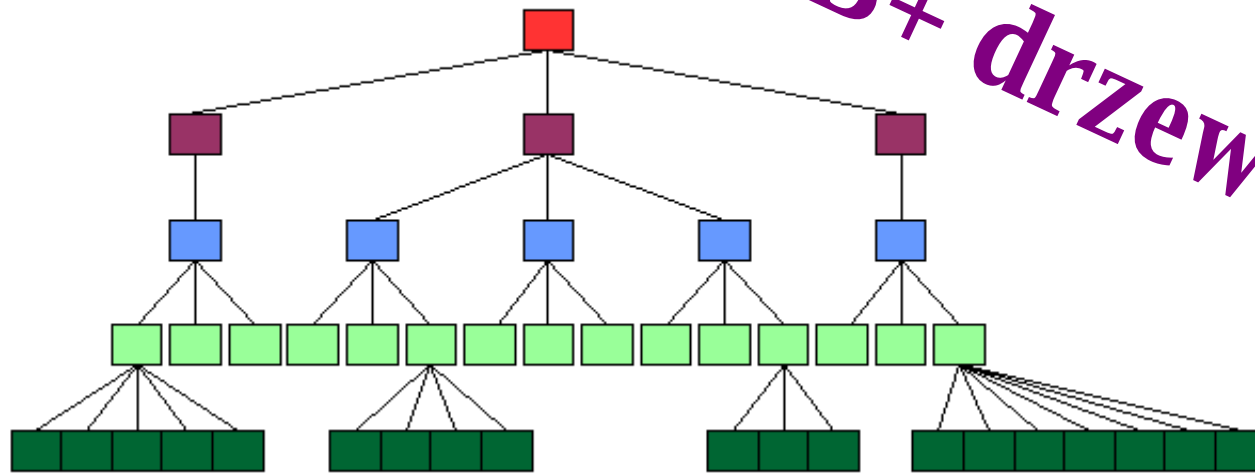
# ReiserFS i Reiser 4 FS

## ReiserFS

- SuSE, Lindows i Gentoo, (Windows? [<http://rfsd.sourceforge.net>])
- Podstawowe cechy ReiserFS to:
  - ✓ **fast journaling** – żeby zminimalizować czas sprawdzania integralności dysku
  - ✓ **używanie szybkich drzew zbalansowanych** – żeby poprawić efektywność dla katalogów zawierających bardzo dużo plików
  - ✓ **efektywne wykorzystywanie miejsca** - “upychanie” wielu małych plików w jednym bloku

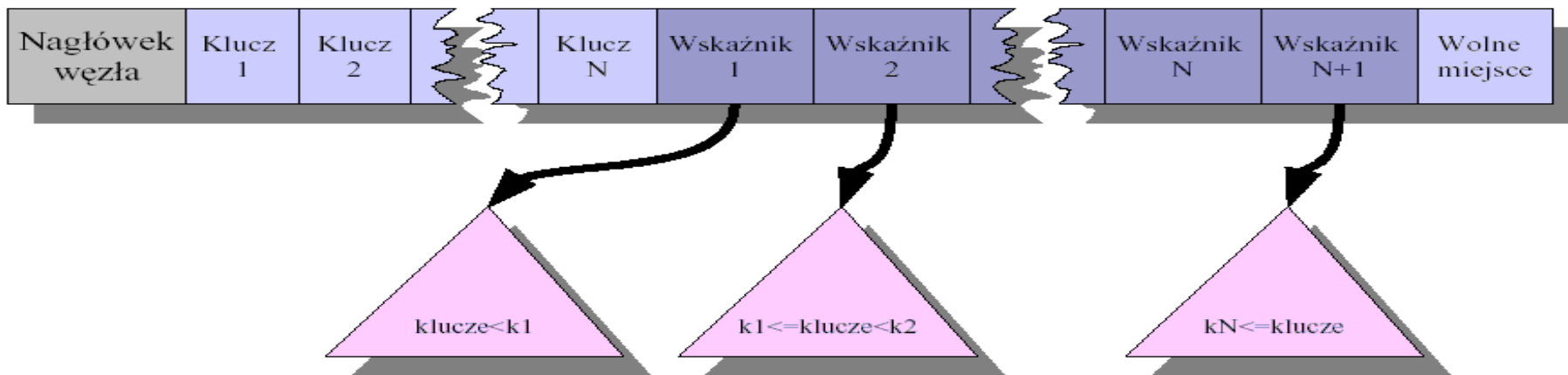


- Level 4**  
Root node
- Level 3**  
Branch nodes
- Level 2**  
Twig nodes
- Level 1**  
Leaf nodes
- BLOBs**

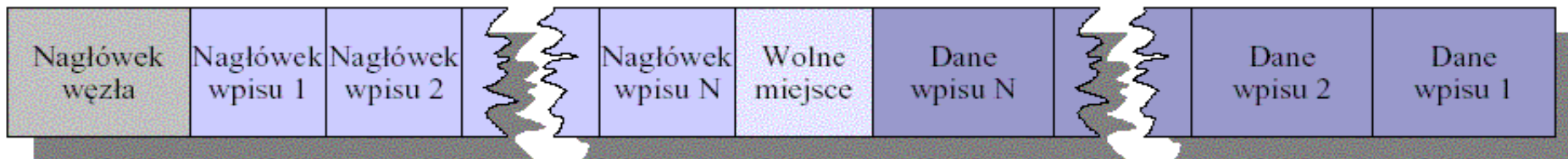


**B+ drzewo**

### Węzeł wewnętrzny



### Węzeł sformatowany



# Nagłówek węzła

Field Name	Type	Size (bytes)	Description
blk_level	unsigned short	2	Level of block in the tree (1-leaf; 2,3,4,...-internal)
blk_nr_item	unsigned short	2	Number of Keys in an Internal block. Or Number of items in a leaf block.
blk_free_space	unsigned short	2	Block Free Space in bytes
blk_right_delim_key	struct key	16	Right delimiting key for this block. (for Leaf nodes only)
	total	6 (8)	(6) 8 bytes for internal nodes; (2) 24 bytes for leaf nodes
		22 (24)	

# Klucz

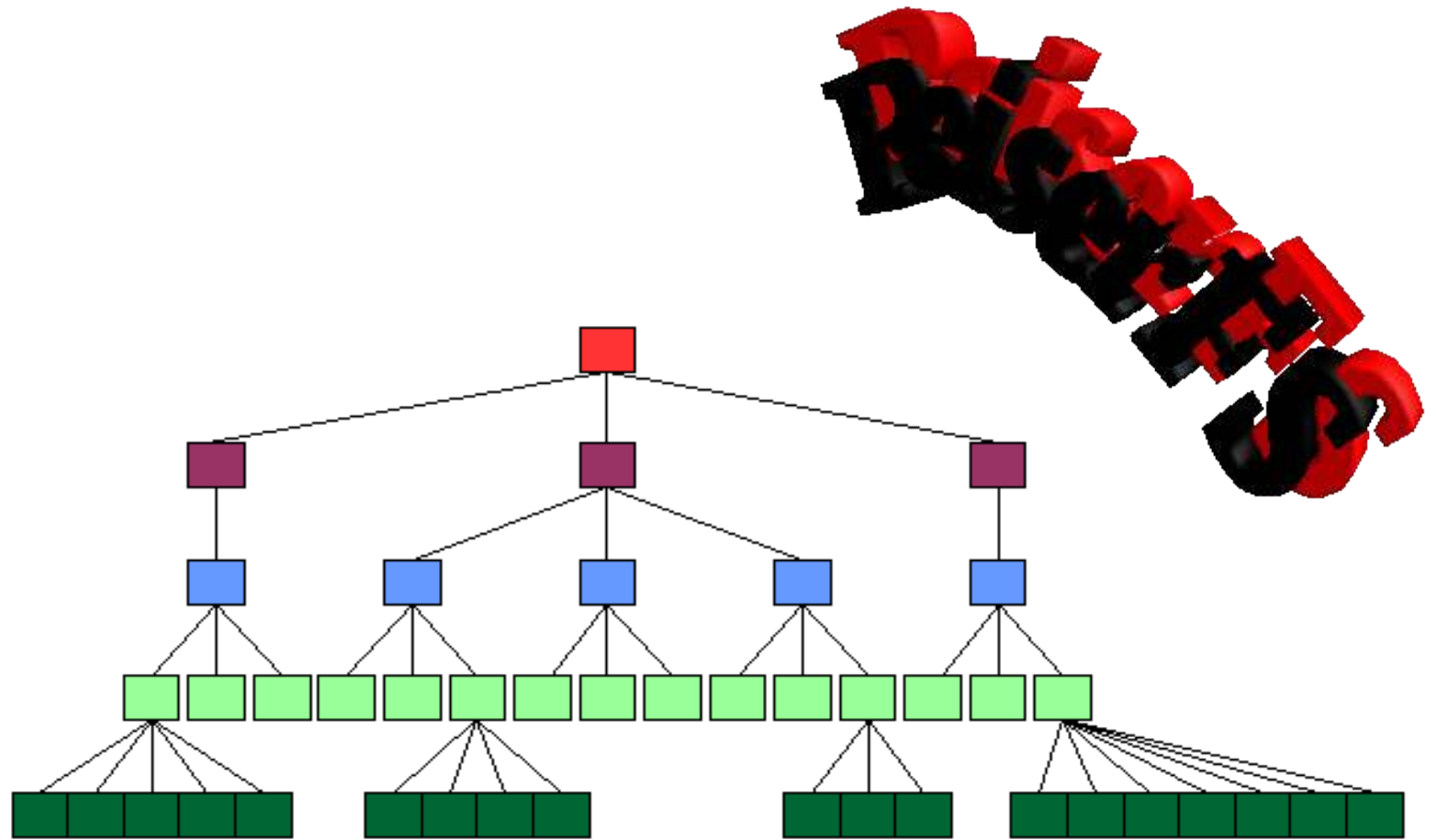
Field Name	Type	Size (bytes)	Description
k_dir_id	u32	4	ID of the parent directory
k_object_id	__u32	4	ID of the object (also it is the number of inode)
k_offset	__u32	4	Offset from beginning of the object to the current byte of the object
k_uniqueness	__u32	4	Type of the item (StatData = 0, Direct = -1, Indirect = -2, Directory = 200)
	total	16	16 bytes

# Wskaźnik

Field Name	Type	Size (bytes)	Description
dc_block_number	unsigned long	4	Disk child's block number.
dc_size	unsigned short	2	Disk child's used space.
	total	6	(6) 8 bytes

# B + d r e w O

- Level 4  
Root node
- Level 3  
Branch nodes
- Level 2  
Twig nodes
- Level 1  
Leaf nodes
- BLOBs





# LEARNING KNOWLEDGE ZONE

Level 4

Root node

Level 3

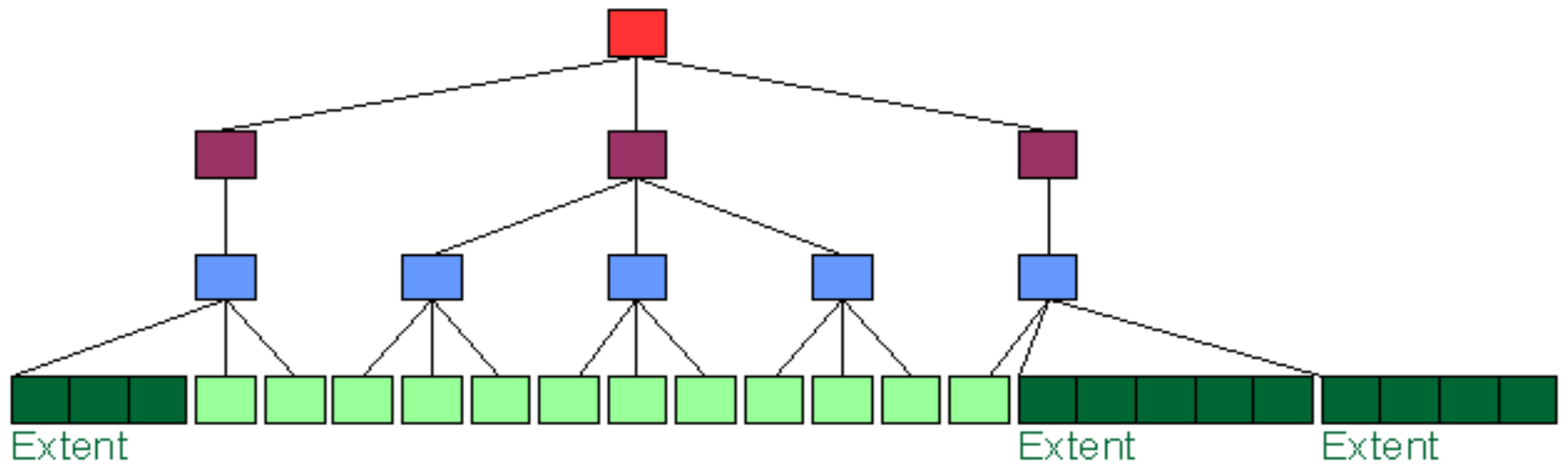
Branch nodes

Level 2

Twig nodes

Level 1

Leaf nodes

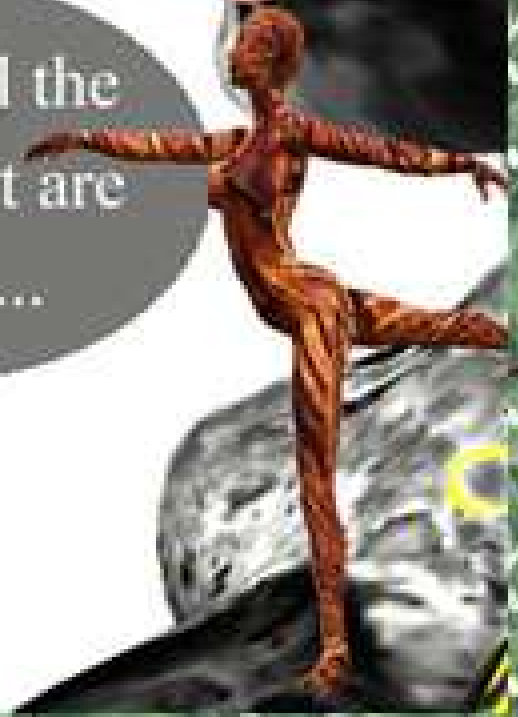




With a dancing  
tree structure, I can just  
shove everything as far  
to the left as possible



freeing all the  
nodes that are  
empty....



Level 4

Root node

Level 3

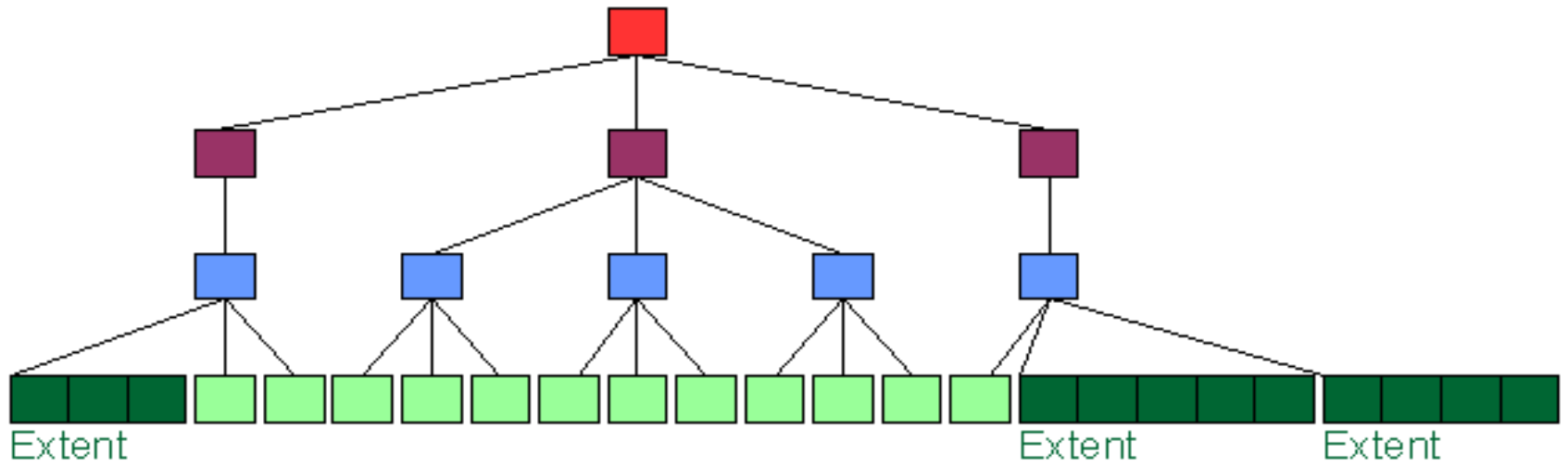
Branch nodes

Level 2

Twig nodes

Level 1

Leaf nodes



*Extent  $i$*



- *file plugins* - udostępniają metody dostępu do plików
- *directory plugins* - udostępniają metody dostępu do plików
- *hash plugins* - obsługują haszowanie kluczy w drzewie tańczącym
- *security plugins* - obsługują bezpieczeństwo
- *item plugins* - udostępniają metody dostępu do pozycji
- *key assignment plugins* - zajmują się przydzielaniem kluczy w drzewie
- *node search and item search plugins* - odpowiedzialne za wyszukiwanie w drzewie węzłów i pozycji



extensibility  
through plugins.....

# **ROZWIĄZANIA**

## **Wersje 5 i 6**

O tym, która wersja, 5 czy 6 zostanie zaimplementowana jako pierwsza mają zdecydować sponsorzy.

## **Wersja 5**

Ma być rozproszonym systemem plików. Niestety w chwili obecnej brak jest dokładnych szczegółów na temat funkcjonalności tego systemu.

## **Wersja 6**

Ma być systemem opartym na rozszerzonej semantyce, z której szczegółami można się zapoznać na stronie Reiser'a.

Główną zmianą w tym systemie ma być odejście od typowo hierarchicznej, drzewiastej struktury katalogów na rzecz dowolnych grafów. Takie podejście spowoduje oczywiście spadek wydajności, lecz według twórców umożliwi, wraz z pewnymi dodatkowymi rozszerzeniami, osiągnięcie znacznie bardziej intuicyjnej i precyzyjnej reprezentacji danych.



**Dziękuję za uwagę**  
**KONIEC**