

# Dostęp do repozytoriów DAV oraz SVN

3 marca 2019



Narodowe Centrum Badań Jądrowych  
<http://www.ncbj.gov.pl/>

# Spis treści

<b>1</b>	<b>Wprowadzenie</b>	<b>3</b>
1.1	Tryby dostępu . . . . .	3
1.2	Autoryzacja użytkowników . . . . .	4
1.3	Przykłady i dane dostępowe . . . . .	4
<b>2</b>	<b>Dostęp w trybie auto-wersjonowania (DAV)</b>	<b>4</b>
2.1	Wybrane narzędzia dostępu w trybie auto-wersjonowania . . . . .	4
2.1.1	CarotDAV . . . . .	5
2.1.2	WinSCP . . . . .	9
2.1.3	CloudSafe . . . . .	13
2.1.4	Davfs2 . . . . .	17
<b>3</b>	<b>Dostęp do repozytorium w trybie pełnego wersjonowania (SVN)</b>	<b>18</b>
3.1	Podstawowe pojęcia i operacje . . . . .	18
3.2	Narzędzia dostępu do repozytorium w trybie pełnego wersjonowania (SVN) . . . . .	21
3.2.1	SmartSVN . . . . .	21
3.2.2	TortoiseSVN . . . . .	35
3.2.3	Polecenie svn w systemach operacyjnych typu POSIX . . . . .	42

# 1 Wprowadzenie



## Subversion

System kontroli wersji **Subversion** (SVN) pozwala przechowywać pliki w centralnej bazie danych, zwanej repozytorium, z zachowaniem pełnej historii zmian, w tym dodawania, usuwania oraz modyfikacji dowolnych plików w repozytorium. W tym dokumencie opisany będzie dostęp do repozytoriów poprzez serwer WWW za pomocą protokołu HTTPS - HTTP z szyfrowaniem SSL które zapewnia bezpieczeństwo i poufność. Inne możliwe protokoły dostępu do repozytoriów, jak np. *svn+ssh* nie będą rozpatrywane.

### Dodatkowe informacje o systemie kontroli wersji Subversion:

- <http://subversion.apache.org/>
- <http://svnbook.red-bean.com/>
- <https://pl.wikipedia.org/wiki/Subversion>



## WebDAV

WebDAV (ang. Web-based Distributed Authoring and Versioning) to rozszerzenie protokołu HTTP umożliwiające zarządzanie plikami znajdującymi się na serwerze.

Z uwagi iż dostęp do repozytoriów SVN protokołem HTTP/HTTPS jest realizowany poprzez moduł DAV, możliwe jest podłączanie repozytoriów SVN jako udziałów WebDAV, co zapewnia dostęp do zawartości repozytorium, jak do zawartości dysku sieciowego.

### Więcej na temat WebDAV można znaleźć tu:

- <http://www.webdav.org/>
- [https://pl.wikipedia.org/wiki/Web-based\\_Distributed\\_Authoring\\_and\\_Versioning](https://pl.wikipedia.org/wiki/Web-based_Distributed_Authoring_and_Versioning)

## 1.1 Tryby dostępu

W tym dokumencie opisane będą dwie metody dostępu:

- Uproszczony dostęp w trybie auto-wersjonowania poprzez protokół WebDAV (metoda prostsza)
- Dostęp w trybie pełnego wersjonowania SVN (metoda zaawansowana)

Dla każdego repozytorium skonfigurowanego na serwerze **uran.ncbj.gov.pl** obydwie metody mogą być stosowane wymiennie. Dostęp do repozytoriów jest możliwy dla wielu użytkowników jednocześnie, jak również jeden użytkownik może mieć w tym samym momencie wiele połączeń z jednym repozytorium, za pomocą różnych programów oraz w różnych trybów dostępu.

## 1.2 Autoryzacja użytkowników

g Dostęp do wybranego repozytorium wymaga autoryzacji:

- Użytkownicy **będący pracownikami NCBJ** logują się za pomocą swojego konta w domenie **CENTRUM**
- Użytkownicy **nie będący pracownikami NCBJ** logują się za pomocą konta stworzonego w wybranej instalacji aplikacji **Redmine**, która jest dedykowana wybranemu projektowi i powiązana z wybranym repozytorium. W takim przypadku wybrana aplikacja Redmine umożliwia zmianę hasła.

## 1.3 Przykłady i dane dostępowe

W tym dokumencie w przykładach został użyty adres repozytorium:

- <https://polfel.ncbj.gov.pl/svn/test>

który składa się z internetowego adresu serwera (<https://polfel.ncbj.gov.pl>), oraz ze ścieżki do repozytorium na danym serwerze (`/svn/test`).

W celu podłączenia się do wybranego repozytorium, należy podać właściwy adres danego repozytorium, oraz swój login.

## 2 Dostęp w trybie auto-wersjonowania (DAV)

W tym trybie dostęp do repozytorium działa jak dostęp do zwykłego katalogu z plikami, takiego jak np. FTP. Użytkownik nie widzi opcji wersjonowania, kontrola wersji odbywa się automatycznie - każda modyfikacja (edycja, dodanie, skasowanie) dowolnego pliku powoduje zwiększenie numeru rewizji repozytorium o 1.

Zaletą tego rozwiązania jest prostota, użytkownik pracuje z plikami jak by były na dysku, a w razie potrzeby automatyczne wersjonowanie pozwala odtworzyć dowolną historyczną wersję każdego pliku. Wadą tego rozwiązania jest dość szybkie zwiększanie numeru rewizji. Dostęp do repozytorium w trybie auto-wersjonowania jest możliwy za pomocą dowolnego klienta obsługującego protokół DAV, poniżej przedstawione są szczegóły konfiguracji wybranych aplikacji.

Korzystanie z auto-wersjonowania nie ogranicza możliwości wersjonowania plików, w każdym przypadku zachowana jest pełna historia zmian. By móc skorzystać z możliwości jakie dostarcza wersjonowanie plików, np. odtworzyć wersję historyczną wybranego pliku, należy użyć aplikacji do dostępu do repozytorium w trybie pełnego wersjonowania (p. 3).

### 2.1 Wybrane narzędzia dostępu w trybie auto-wersjonowania

W tej sekcji przedstawione zostaną następujące programy:

- CarotDAV (p. 2.1.1)
- WinSCP (p. 2.1.2)
- CloudSafe (p. 2.1.3)
- Davfs2 (p. 2.1.4)

Do korzystania z udziałów WebDAV (repozytoriów SVN w trybie auto-wersjonowania), wystarczy **jeden** wybrany program. W tym dokumencie opisano korzystanie z różnych aplikacji by pozostawić użytkownikowi możliwość wyboru na wypadek gdyby np. z użyciem któregoś programu występowały problemy.

### 2.1.1 CarotDAV

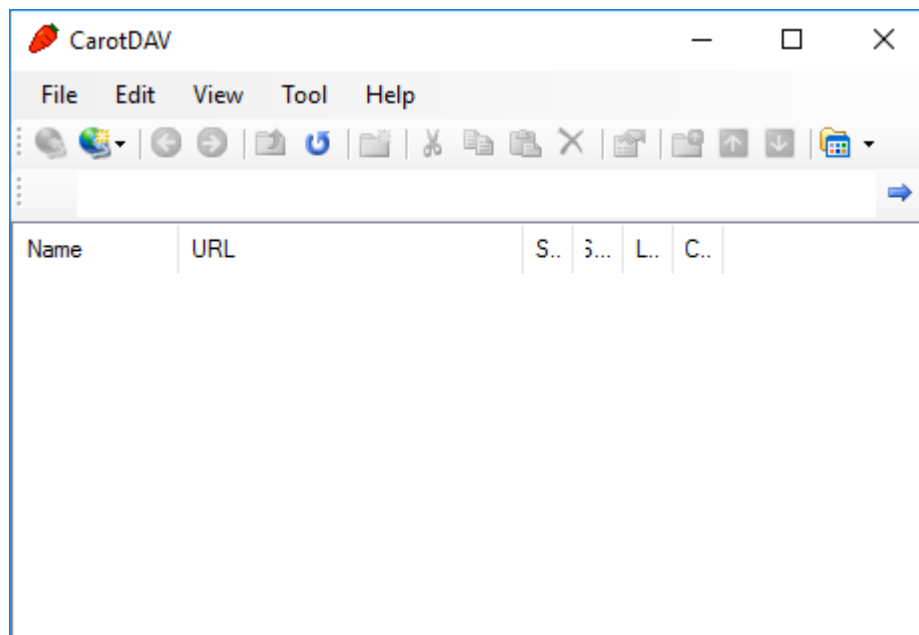
Program CarotDAV umożliwia dostęp do zdalnych repozytoriów m.in. za pomocą protokołu WebDAV. Jest to najprostszy z programów z opisanych w tym dokumencie, i z tego względu jest on rekomendowany dla osób które chcą korzystać z transferu plików bez konieczności zgłębiania szczegółów kontroli wersji.

#### Zasoby

- [Strona domowa projektu CarotDAV](#)
- [CarotDAV1.15.5.zip](#) - wersja instalacyjna
- [CarotDAV1.15.5.portable.zip](#) - wersja przenośna nie wymagająca instalacji

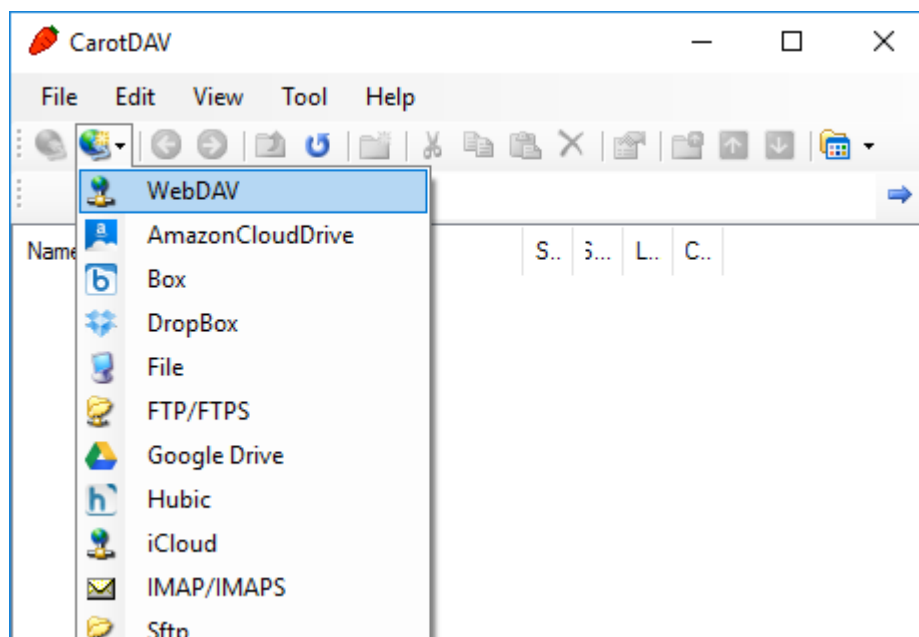
#### Instalacja i uruchomienie

Program CarotDAV należy zainstalować lub rozpakować, zależnie od wybranej opcji, a następnie uruchomić - rys. 1.



Rysunek 1: Program po instalacji i uruchomieniu

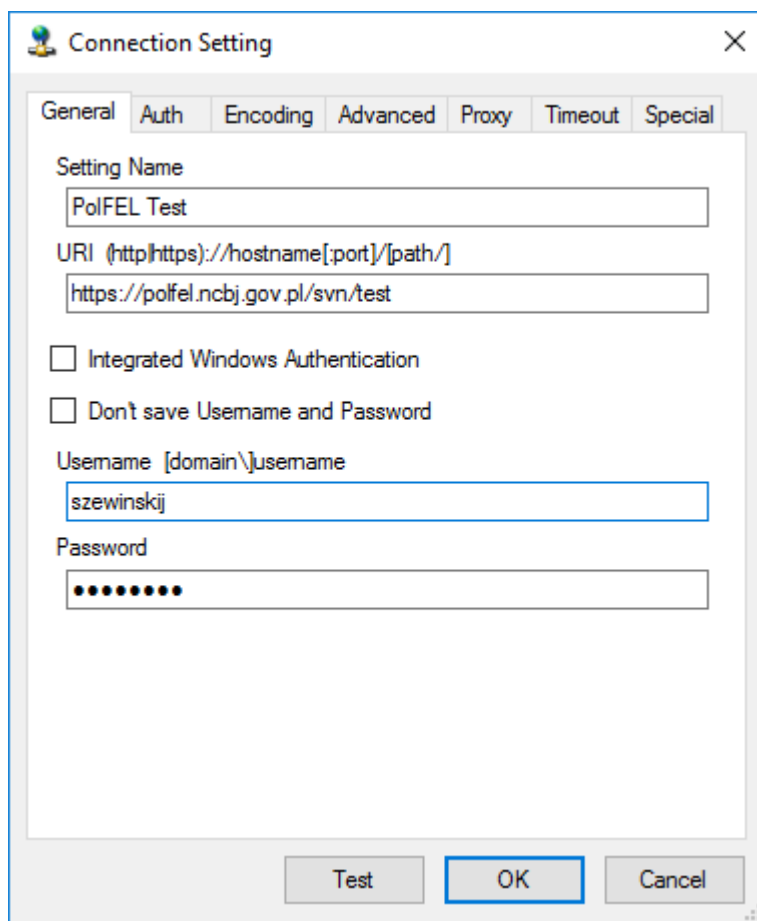
Dla każdego repozytorium należy stworzyć połączenie typu WebDAV (rys. 2):



Rysunek 2: Dodawanie połączenia WebDAV w programie CarotDAV

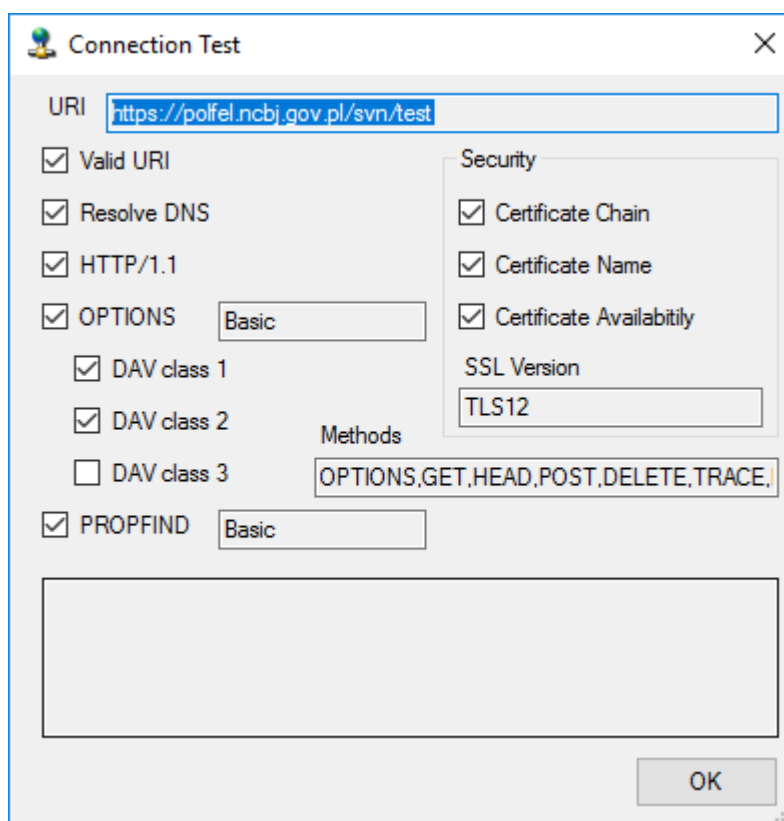
W ustawieniach każdego połączenia należy ustawić parametry, w szczególności (rys. 3):

- URL repozytorium
- Nazwę użytkownika
- Hasło



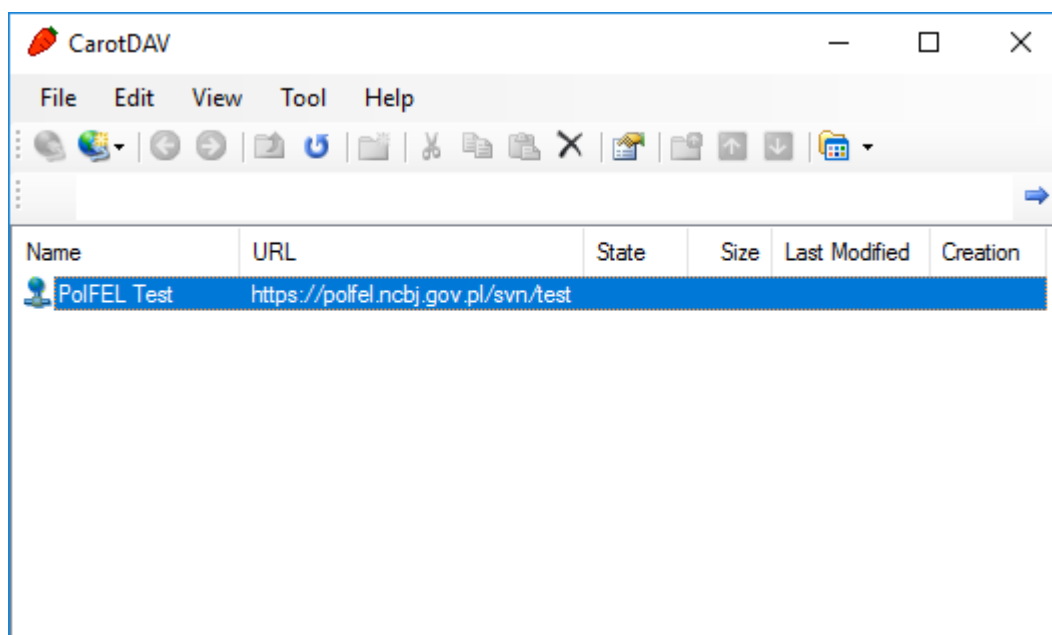
Rysunek 3: Parametry połączenia w programie CarotDAV

Po wpisaniu odpowiednich wartości, opcjonalnie można przetestować poprawność konfiguracji klikając *Test*. Po podaniu poprawnych danych, wynik pomyślnego testu wygląda następująco (rys. 4):



Rysunek 4: Test poprawnie skonfigurowanego połączenia WebDAV

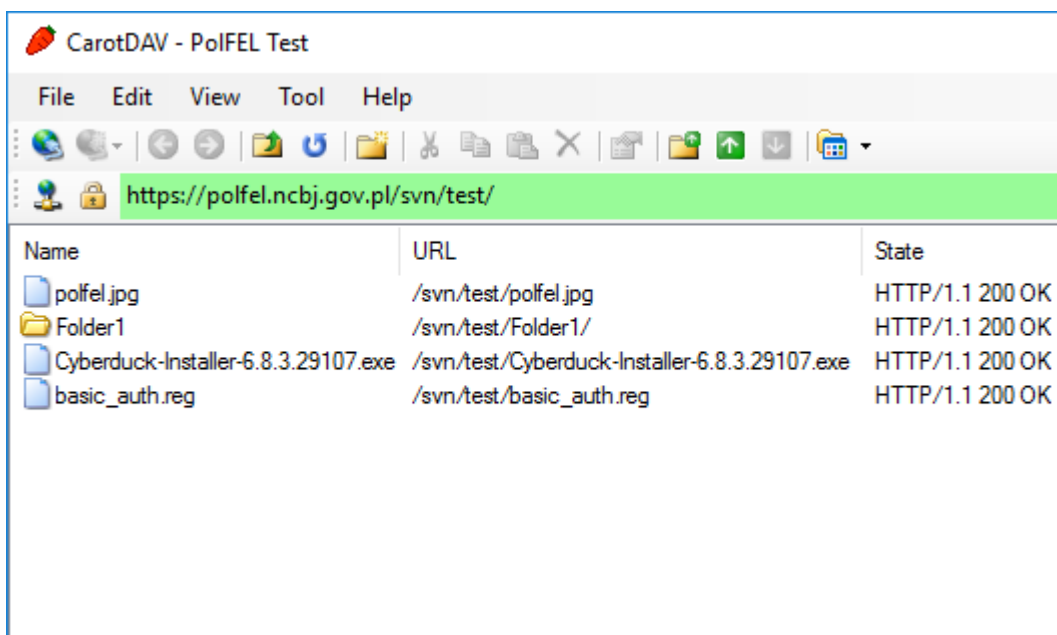
Nowe połączenie będzie dostępne na liście połączeń (rys. 5):



Rysunek 5: Lista połączeń w programie CarotDAV

Po otwarciu utworzonego połączenia, można kopiować, przenosić i usuwać pliki w repozytorium (rys. 6):





Rysunek 6: Zawartość repozytorium dostępna w programie CarotDAV

### 2.1.2 WinSCP

WinSCP umożliwia przesyłanie plików różnymi protokołami, m.in. za pomocą WebDAV.

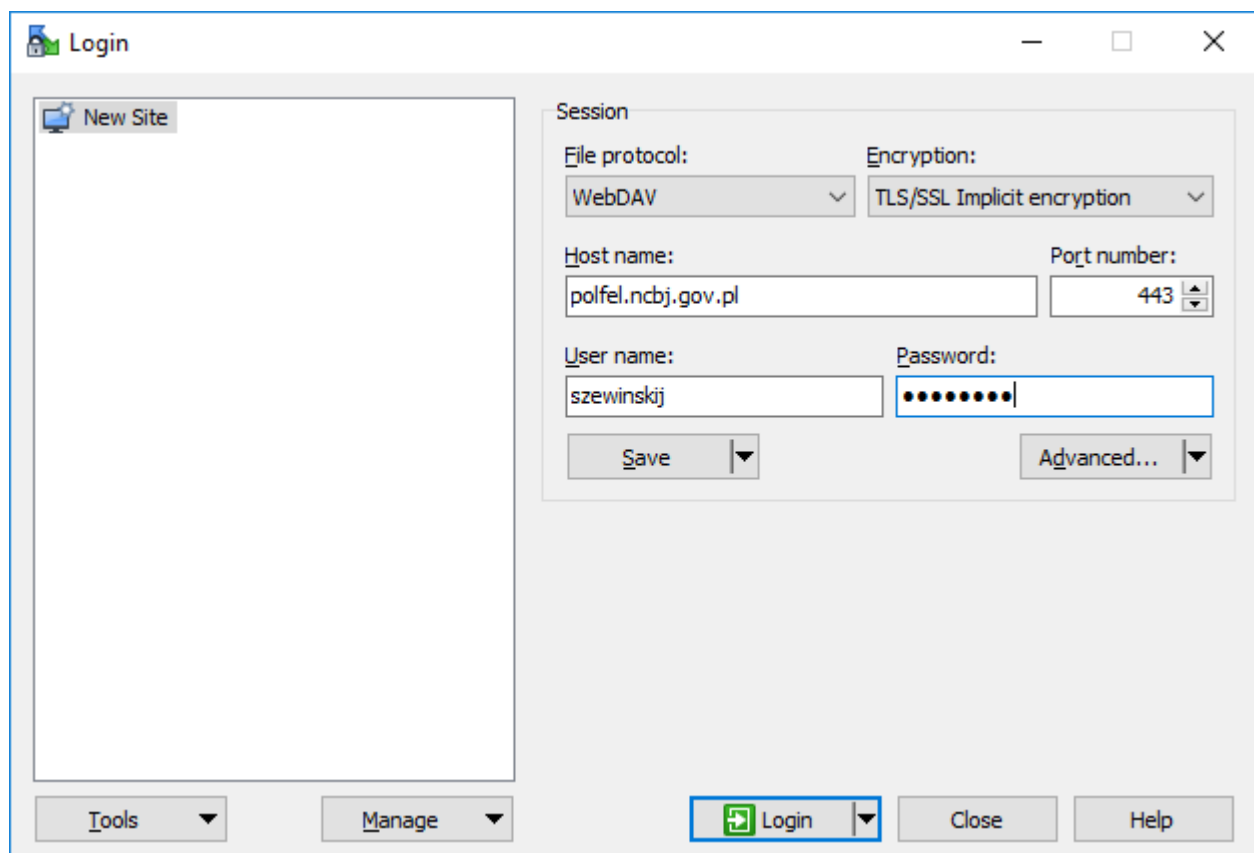
#### Zasoby

- [Strona domowa programu WinSCP](#)
- [WinSCP-5.13.7-Setup.exe](#) - wersja instalacyjna
- [WinSCP-5.13.7-Portable.zip](#) - wersja przenośna nie wymagająca instalacji

#### Instalacja i uruchomienie

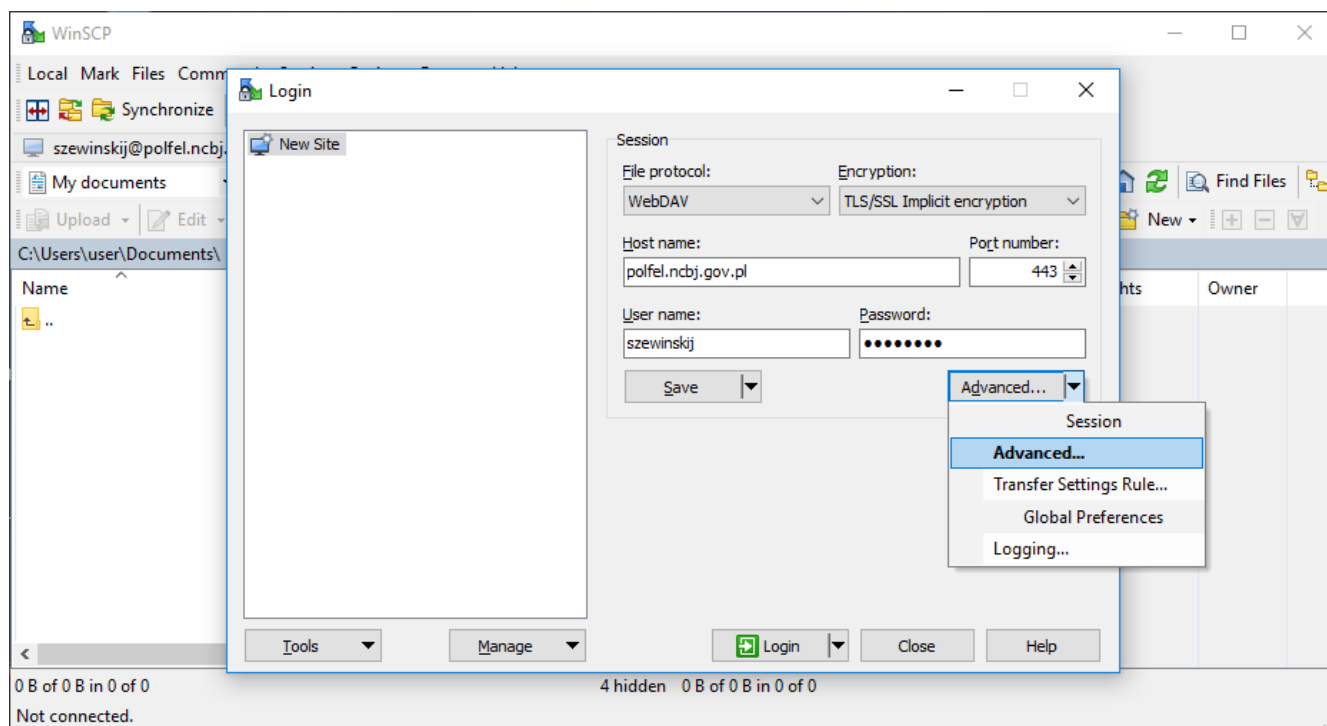
Program należy zainstalować lub rozpakować w przypadku wersji przenośnej. Po uruchomieniu należy wpisać parametry nowej sesji (rys. 7):

- Protokół: **WebDAV**
- Szyfrowanie: **TLS/SSL**
- Adres serwera (bez ścieżki); dla repozytorium <https://polfel.ncbj.gov.pl/svn/test> będzie to **https://polfel.ncbj.gov.pl**
- Port połączenia HTTP SSL: **443**
- nazwę użytkownika
- hasło



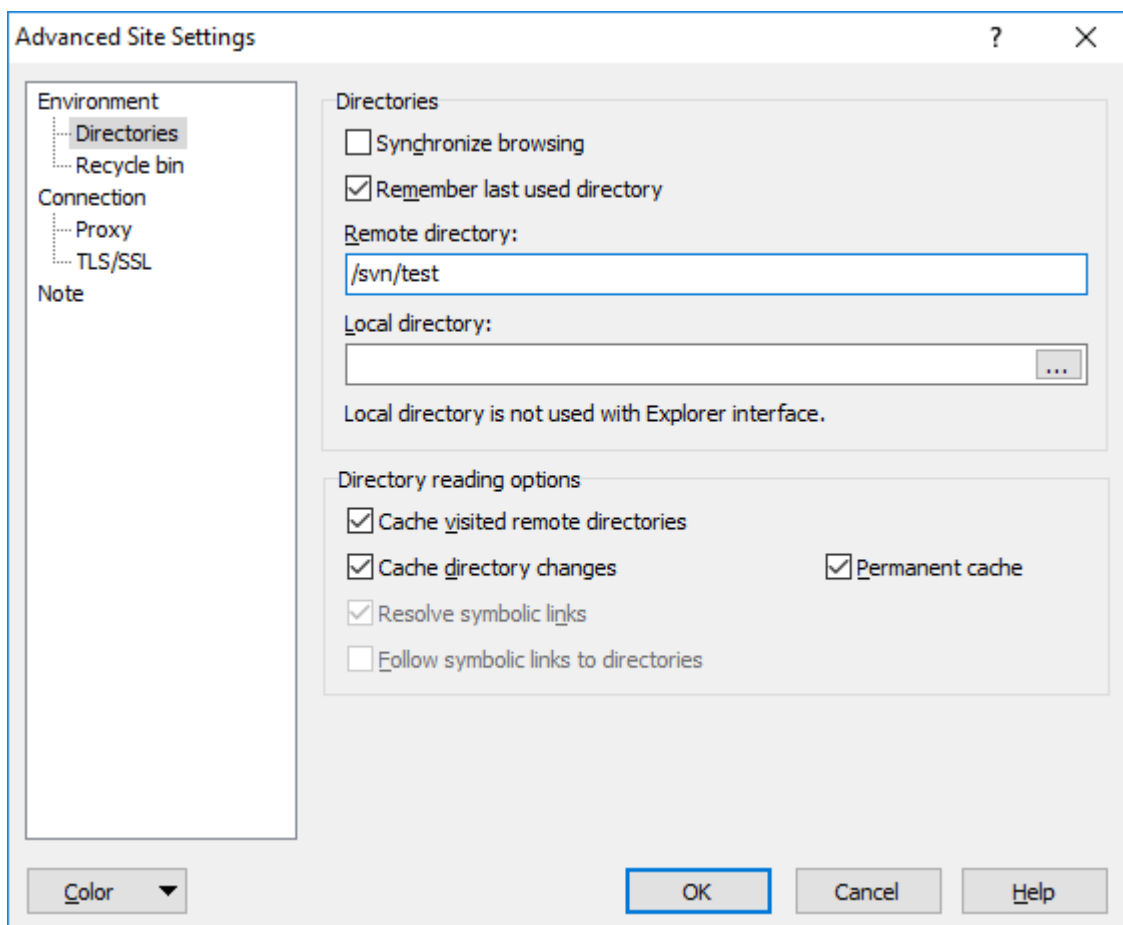
Rysunek 7: Dodawanie nowego połączenia w programie WinSCP

W celu poprawnej konfiguracji połączenia, należy wybrać opcje zaawansowane (rys. 8)



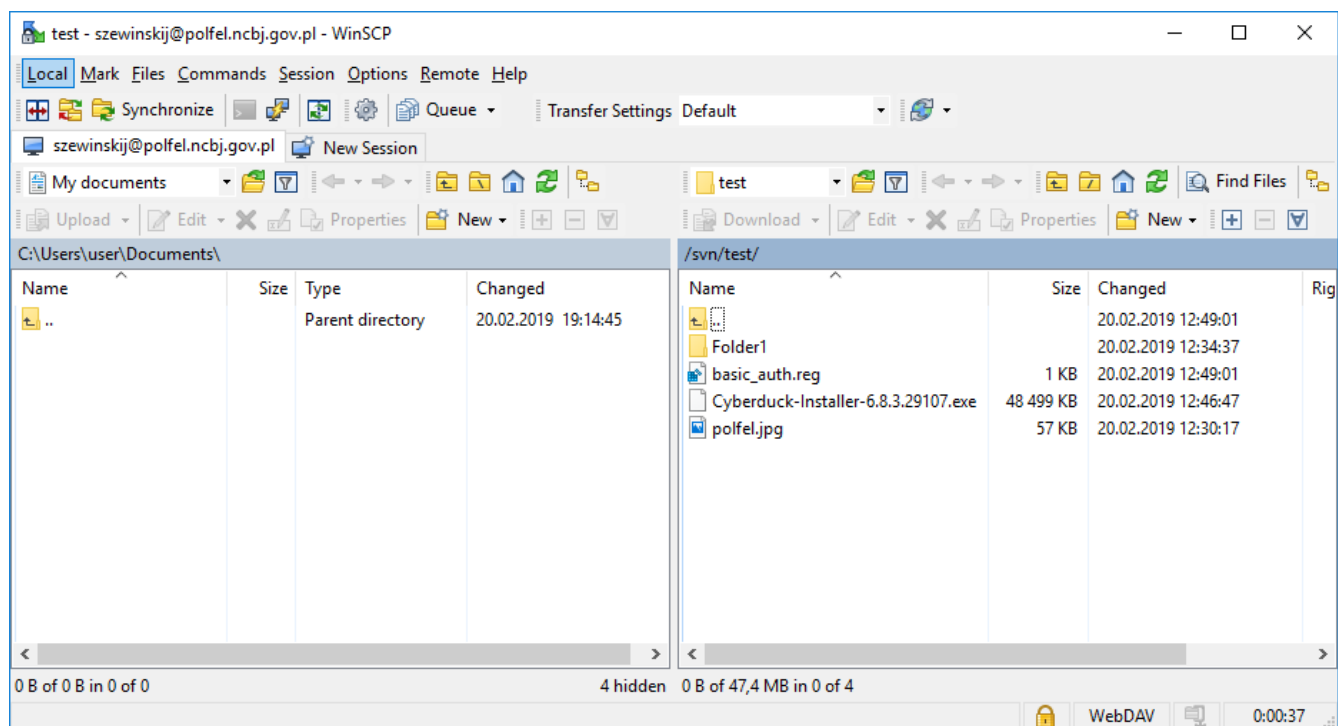
Rysunek 8: Wybór zaawansowanych opcji połączenia w programie WinSCP

W opcjach zaawansowanych należy ustawić **ścieżkę** repozytorium na serwerze. Dla repozytorium <https://polfel.ncbj.gov.pl/svn/test> będzie to `/svn/test` (rys. 9)



Rysunek 9: Zaawansowane opcje połączenia w programie WinSCP

Po nawiązaniu połączenia można uzyskać dostęp do plików w repozytorium (rys. 10):



Rysunek 10: Zawartość repozytorium dostępna w programie WinSCP

### 2.1.3 CloudSafe

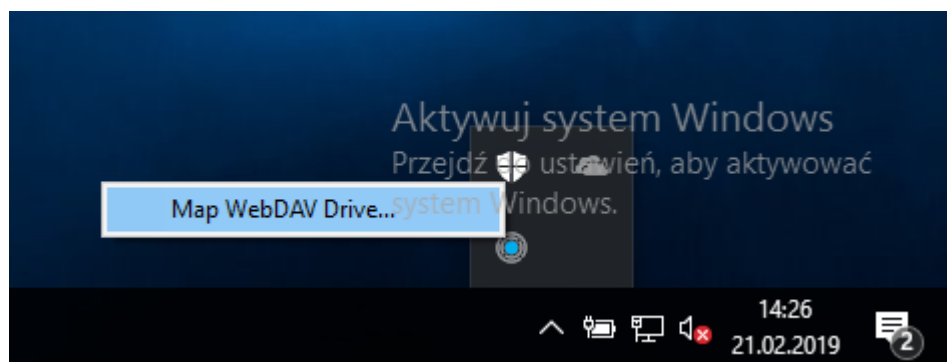
Program CloudSafe umożliwia zamontowanie zdalnego repozytorium DAV jako dysku sieciowego w systemie MS Windows. W odróżnieniu od poprzednich rozwiązań, CloudSafe umożliwia otwarcie i edycję plików bezpośrednio z repozytorium bez konieczności ich uprzedniego kopiowania na dysk lokalny. Zaletą tego rozwiązania jest prostota i oszczędność miejsca na dysku - nie trzeba kopiować całego (lub części) repozytorium. Wadą jest to że w momencie zerwania połączenia z siecią komputerową (z internetem), może nie być możliwe zapisanie zmian w edytowanych plikach. Należy też pamiętać że każde zapisanie pliku otwartego bezpośrednio z repozytorium wiąże się ze zwiększeniem numeru rewizji o 1.

#### Zasoby

- [CloudSafeWebDAVDrive\\_x86\\_1.2.758.msi](#) - wersja 32-bitowa
- [CloudSafeWebDAVDrive\\_x64\\_1.2.758.msi](#) - wersja 64-bitowa

#### Instalacja i uruchomienie

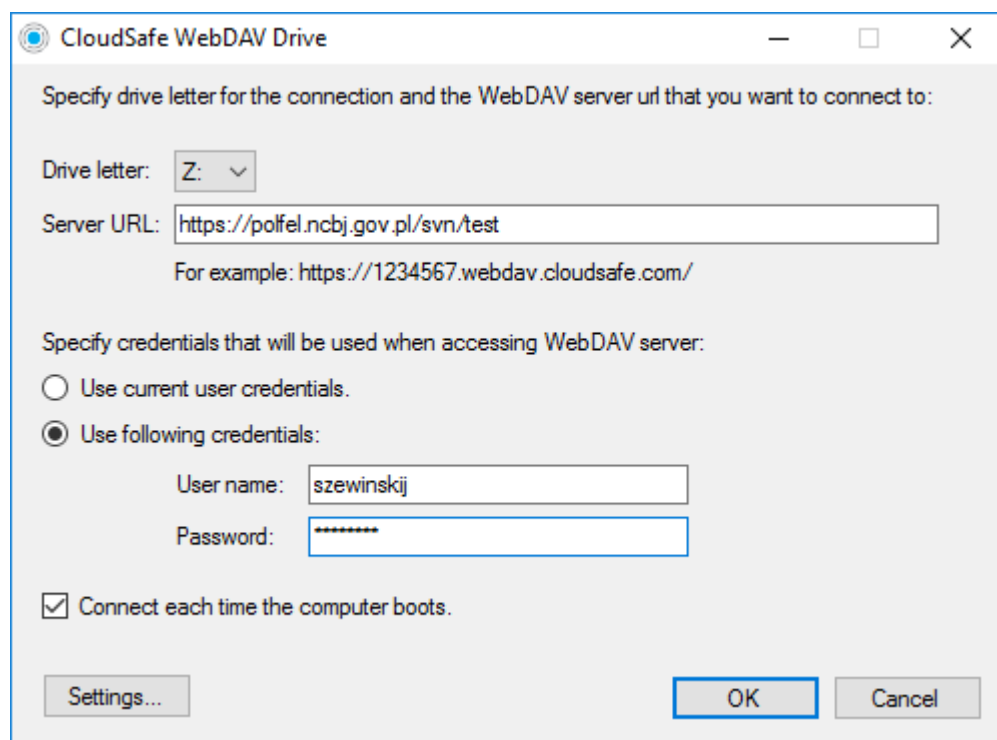
Po zainstalowaniu i uruchomieniu, ikona programu powinna być dostępna w obszarze powiadomień systemu Windows - rys. 11. By dodać nowe połączenie do udziału DAV, należy kliknąć tą ikonę prawym guzikiem myszy i wybrać jedyną dostępną opcję "Map WebDAV Drive...".



Rysunek 11: Dodawanie nowego połączenia z dyskiem sieciowym WebDAV

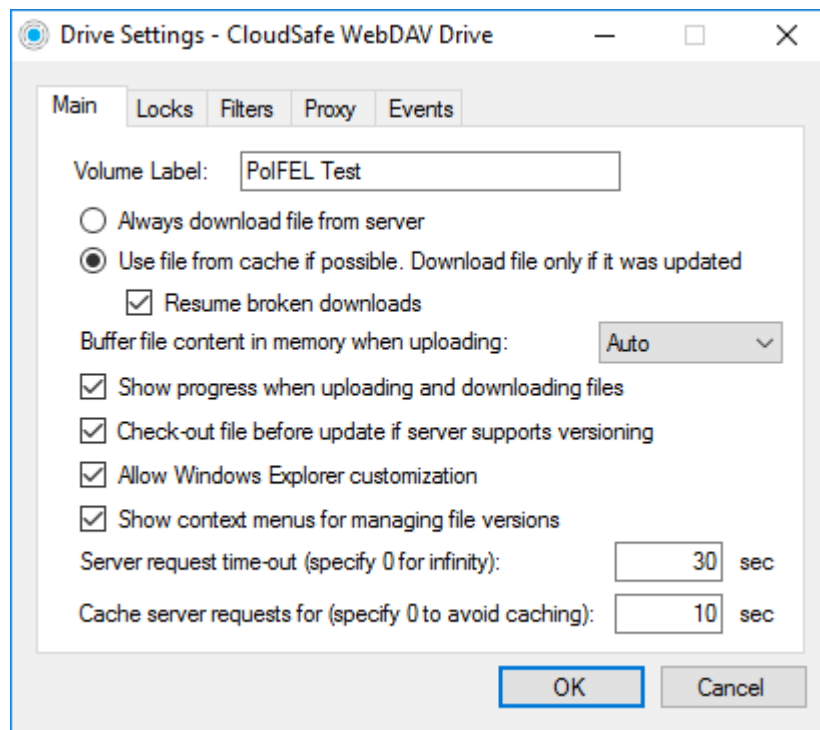
W oknie dodawania nowego połączenia (rys. 12) należy podać:

- URL repozytorium (adres serwera wraz ze ścieżką do wybranego repozytorium)
- Nazwę użytkownika
- Hasło



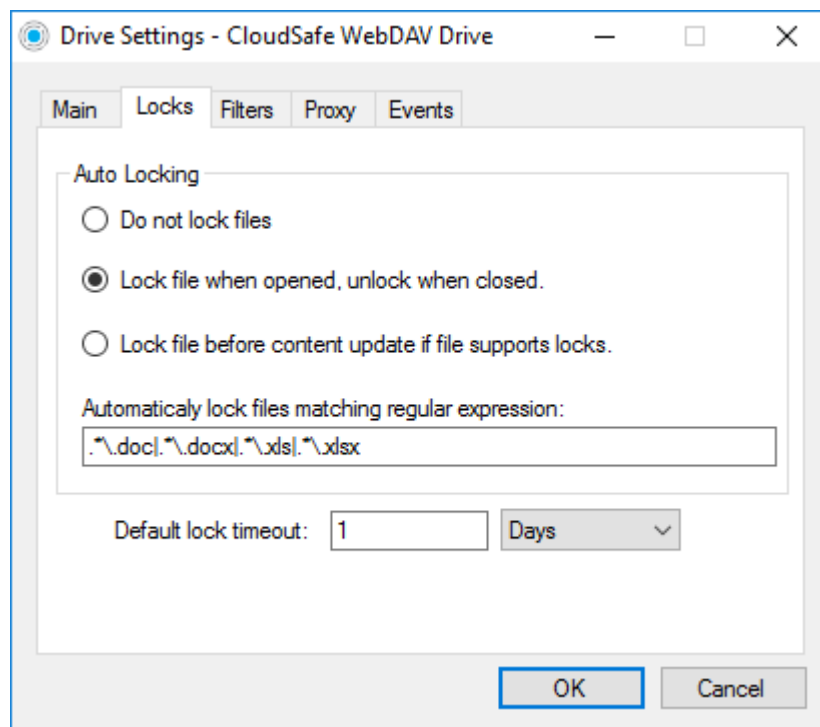
Rysunek 12: Okno ustawień nowego połączenia z dyskiem WebDAV

Nim się utworzy nowe połączenie, warto przejrzeć i dostosować do własnych potrzeb ustawienia zaawansowane klikając *Settings* (rys. 13):



Rysunek 13: Opcje zaawansowane połączenia z dyskiem WebDAV w programie CloudSafe

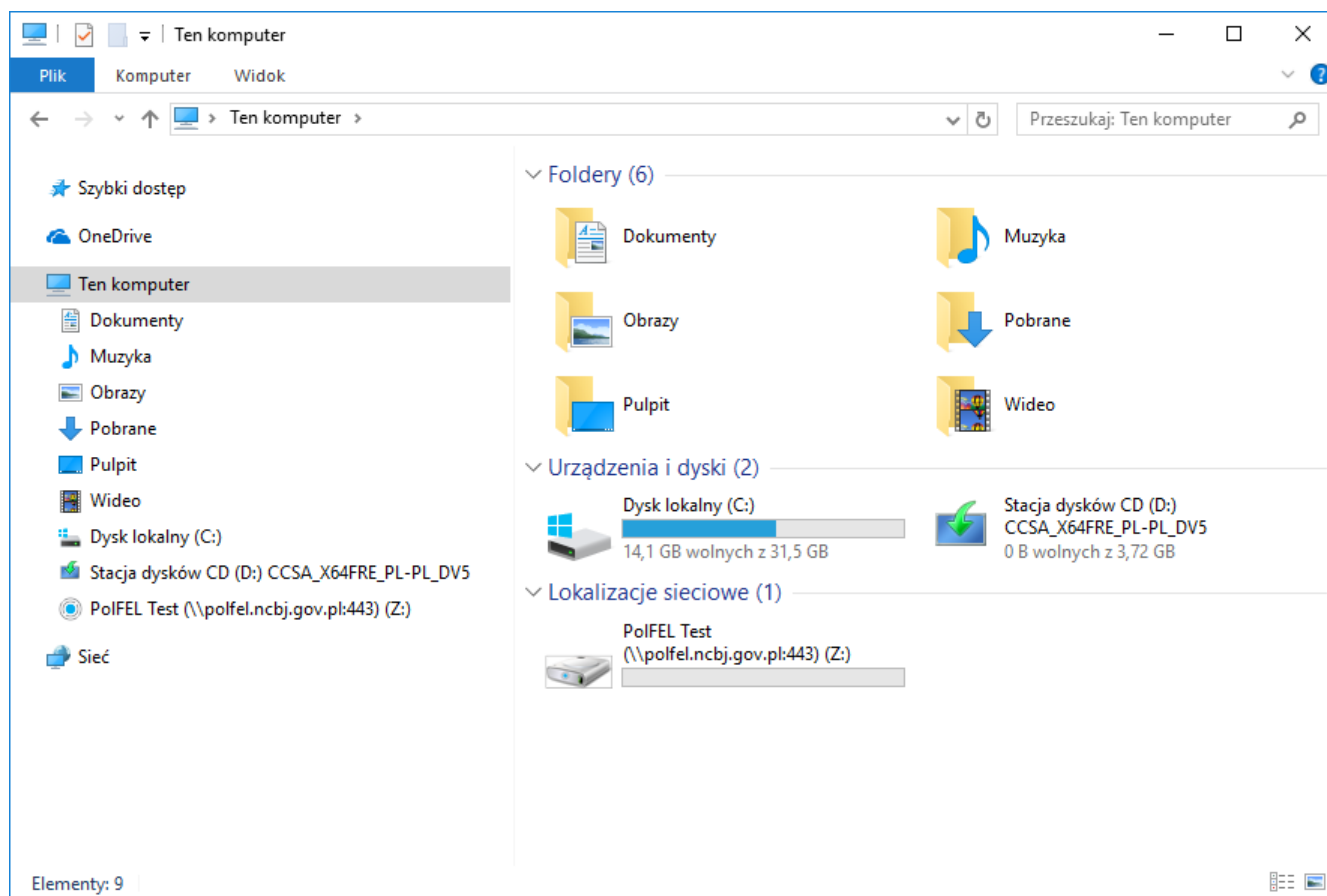
W szczególności warto rozważyć możliwość blokowania i zapewnienia dostępu na wyłączność do plików otwieranych do edycji bezpośrednio z repozytorium (rys. 14):



Rysunek 14: Opcje blokowania dostępu do plików w programie CloudSafe

Po utworzeniu połączenia, nowy dysk sieciowy WebDAV jest dostępny w oknie "Ten komputer"

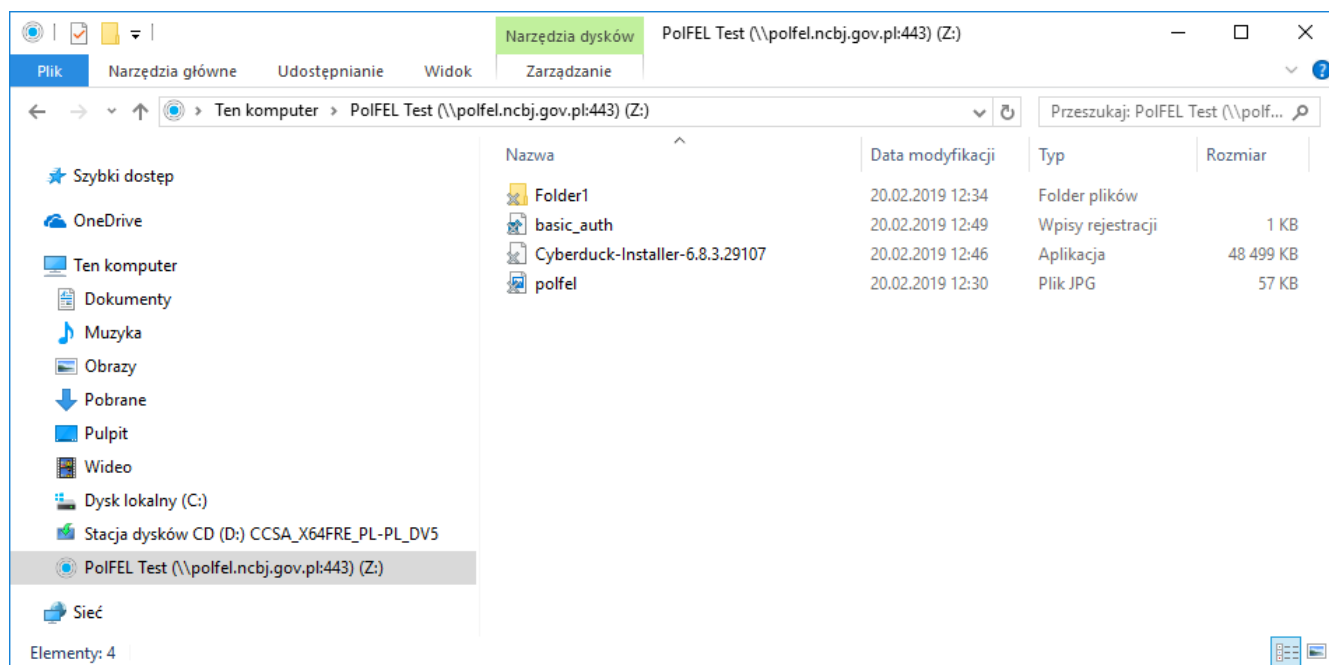
Systemu MS Windows (rys. 15)



Rysunek 15: Nowy dysk sieciowy dostępny w oknie "Ten komputer"

Zawartość repozytorium jest dostępna tak jak zawartość dysku lokalnego (rys. 16).





Rysunek 16: Dostęp do zawartości repozytorium poprzez dysk sieciowy CloudSafe WebDAV

## 2.1.4 Davfs2

W systemach operacyjnych zgodnych ze standardem POSIX, takich jak np. Linux, dzięki oprogramowaniu [davfs2](#) możliwe jest zamontowanie zasobów WebDAV tak jak zwykłego systemu plików.

### Zasoby

- Strona domowa: <http://savannah.nongnu.org/projects/davfs2>
- Strona Wiki: <https://en.wikipedia.org/wiki/Davfs2>

### Instalacja i uruchomienie

Instalacja jest zależna od dystrybucji systemu operacyjnego, przykłady użycia są przedstawione poniżej:

Montowanie Udziały WebDAV:

```
root@carbon:~# mount -t davfs https://polfel.ncbj.gov.pl/svn/test ~/mnt
Please enter the username to authenticate with server
https://polfel.ncbj.gov.pl/svn/test or hit enter for none.
Username: szewinskij
Please enter the password to authenticate user szewinskij with server
https://polfel.ncbj.gov.pl/svn/test or hit enter for none.
Password:
```

Sprawdzenie czy udział WebDAV został zamontowany:

```

root@carbon:~# mount
/dev/nvme0n1p5 on / type ext4 (rw)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
tmpfs on /dev/shm type tmpfs (rw)
/dev/nvme0n1p1 on /boot/efi type vfat (rw,noatime)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
https://polfel.ncbj.gov.pl/svn/test on /root/mnt type davfs (rw,nosuid,nodev,_netdev,uid=0,

```

Wyświetlenie zawartości zamontowanego udziału:

```

root@carbon:~# ls -la ~/mnt
total 48556
drwxr-xr-x 6 root root      344 Feb 22 13:24 ./
-rw-r--r-- 1 root root 49662224 Feb 20 12:46 Cyberduck-Installer-6.8.3.29107.exe
drwxr-xr-x 2 root root        0 Feb 20 12:34 Folder1/
drwxr-xr-x 2 root root        0 Feb 21 15:10 System\ Volume\ Information/
-rw-r--r-- 1 root root      336 Feb 20 12:49 basic_auth.reg
drwx----- 2 root root        0 Mar  1 08:56 lost+found/
-rw-r--r-- 1 root root    57727 Feb 20 12:30 polfel.jpg

```

Odmontowanie udziału i ponowne wyświetlenie zawartości:

```

root@carbon:~# umount ~/mnt
/sbin/umount.davfs: waiting while mount.davfs (pid 25457) synchronizes the cache .. OK
root@carbon:~# ls -la ~/mnt
total 8
drwxr-xr-x  2 root root 4096 Mar  1 08:55 ./
drwx--x--- 44 root root 4096 Mar  1 08:55 ../
root@carbon:~#

```

### 3 Dostęp do repozytorium w trybie pełnego wersjonowania (SVN)

W tym trybie użytkownik ma pełną kontrolę nad wersjonowaniem plików, decyduje w którym momencie zmiany dokonane w jednym lub w wielu plikach mają zostać wysłane do repozytorium w postaci jednej atomowej operacji *commit*, zwiększając wartość rewizji o 1. Użytkownik ponadto może przeglądać historię zmian wybranych plików, pobierać ich historyczne wersje, porównywać zmiany dokonane lokalnie względem ostatniej rewizji pobranej z repozytorium itp.

Zaletą tego podejścia jest pełna kontrola nad wersjonowaniem plików oraz wynikające z tego usprawnienie pracy z plikami objętymi kontrolą wersji. Wadą tego rozwiązania jest większa złożoność i konieczność przyswojenia dodatkowej wiedzy nt. systemu kontroli wersji Subversion.

#### 3.1 Podstawowe pojęcia i operacje

Wybrane podstawowe pojęcia

- **repozytorium** - baza danych na serwerze posiadająca pełną historię zmian dokonywanych w wersjonowanych plikach
- **kopia robocza** - najnowsza (lub wybrana) wersja plików z repozytorium na komputerze lokalnym. Użytkownik może mieć wiele kopii roboczych związanych z jednym repozytorium.
- **rewizja** - Numer określający wersję plików w repozytorium, całe repozytorium ma jeden numer rewizji, który jest zwiększany o 1 w przypadku pomyślnego zatwierdzenia zmian (operacja *commit*)
- **head** - określenie aktualnie najnowszej rewizji
- **branch** - rozgałęzienie równoległe wersji wybranych plików w repozytorium
- **merge** - złączenie rozgałęzienia do wybranej lub głównej gałęzi
- **tag** - oznaczenie (i zamrożenie) wybranej rewizji (bez planów na dalszy rozwój i złączania z inną gałęzią)
- **trunk** - określenie głównej gałęzi rozwoju
- **konflikt** - sytuacja gdy dwóch użytkowników chce wgrać różne dane jako tą samą rewizję

### Wybrane operacje

- **checkout** - pobranie najnowszej (lub wybranej) wersji danych z repozytorium i utworzenie kopii roboczej - wykonuje się to raz dla każdej kopii roboczej
- **update** - zaktualizowanie kopii roboczej do najnowszej (lub wybranej) rewizji z repozytorium
- **commit** - wysłanie zmian dokonanych w kopii roboczej do repozytorium - to zwiększy wersję repozytorium o jeden. Ta operacja jest **atomowa** - albo wszystkie zmiany są wprowadzane do repozytorium, albo żadne
- **revert** - cofnięcie lokalnych zmian dokonanych w kopii roboczej do ostatniej wersji pobranej z repozytorium (przez **checkout** lub **update**)
- **add** - dodanie niewersjonowanych plików do wersjonowania w ramach kopii roboczej. Nowododane pliki mają status 'dodane' i należy jeszcze wykonać operację **commit** by zostały wysłane do repozytorium i były dostępne dla innych użytkowników
- **del** - usunięcie plików z wersjonowania, to się wiąże (ale nie jest tożsame) ze skasowaniem plików z dysku. Fakt usunięcia pliku z wersjonowania wymaga zatwierdzenia komendą **commit**
- **status** - wyświetlenie stanu plików w kopii roboczej - które są zmodyfikowane, dodane, usunięte, w konflikcie itp.
- **blame/annotate** - wyświetlenie przez kogo i kiedy, poszczególne linie w danym pliku (tekstowym) były modyfikowane
- **diff** - wyświetlenie różnic w pliku (tekstowym) pomiędzy aktualnym (zmodyfikowanym) stanem w kopii roboczej, a referencyjną (najnowszą lub wybraną) wersją w repozytorium

## Typowe wykorzystanie systemu Subversion

By rozpocząć pracę z Subversion (zakładając że repozytorium zostało stworzone), należy wykonać operację `checkout`, w wyniku której zostaną pobrane pliki z repozytorium, i zostanie utworzona lokalna kopia robocza. Operację `checkout` wykonuje się raz dla każdej kopii roboczej.

Typowy scenariusz pracy z plikami pod kontrolą Subversion jest następujący:

1. Przed rozpoczęciem pracy należy wykonać `update` i zaktualizować swoją kopię roboczą - pobrać ew. zmiany które zostały dokonane w repozytorium od czasu ostatniej synchronizacji. Zmiany mogły być dokonane przez innego użytkownika, lub z innego komputera (np. pobranie na komputer w pracy zmian wgranych z komputera domowego).
2. Mając zaktualizowaną kopię roboczą można normalnie pracować i dokonywać zmian w plikach
3. W czasie dokonywania zmian można podglądać co zostało zmodyfikowane poleceniami `status` i `diff` (dla plików tekstowych).
4. W przypadku gdy dokonane zmiany lokalne nie są właściwe, można je wycofać poleceniem `revert`.
5. Jeśli dokonane zmiany są kompletne i sprawdzone, to możemy je przesłać do repozytorium wykonując operację `commit`. Należy pamiętać o ew. dodaniu nowych plików, o ile powstały i powinny być wersjonowane (i o usunięciu tych które nie mają już być obsługiwane). Dobrym zwyczajem jest dodanie krótkiej informacji tekstowej, która opisuje dokonane zmiany (programy do obsługi SVN dają możliwość dodania opisu a niekiedy nawet tego wymagają).

W przypadku gdy ktoś inny pobrał dany plik, przykładowo `plik.txt`, z repozytorium w tej samej rewizji, przykładowo 5, dokonał zmian i przeprowadził operację `commit` pomyślnie tworząc w repozytorium rewizję 6, wówczas próba przeprowadzenia przez nas operacji `commit` zakończy się niepowodzeniem, a po przeprowadzeniu operacji `update` stan ww. pliku zostanie określony jako *Konflikt (C)*.

W takiej sytuacji system Subversion tworzy w kopii roboczej następujące pliki:

- `plik.txt` - plik ze znacznikami do połączenia
- `plik.txt.mine` - plik z naszymi modyfikacjami
- `plik.txt.r5` - plik w wersji bazowej przed dokonaniem naszych lokalnych zmian (w tym przykładzie rewizja 5)
- `plik.txt.r6` - plik w najbardziej aktualnej wersji, jaka została poprawnie zatwierdzona w repozytorium

Znaczniki w pliku wynikowym są w następującym formacie:

```
...  
Wspólna linia  
Wspólna linia  
<<<<<<< .mine
```

```
Zmiany wprowadzone przez nas
...
Zmiany wprowadzone przez nas
=====
Zmiany wprowadzone przez innego użytkownika w rewizji 6
...
Zmiany wprowadzone przez innego użytkownika w rewizji 6
>>>>>>> .r6
Wspólna linia
Wspólna linia
...
```

By rozwiązać konflikt można:

- Analizować każdy obszar w pliku `plik.txt` i decydować którą wersję każdego spornego obszaru wybrać. Opcję edycji i wybierania obszarów spornych ze znacznikami mają wybrane edytory tekstu, np. Emacs.
- Usunąć plik ze znacznikami `plik.txt` i zastąpić go plikiem `plik.txt.mine` wymuszając swoje zmiany
- Zmodyfikować cały projekt tak, żeby funkcjonował poprawnie razem ze zmianami dokonanymi w repozytorium

W przypadku, gdy konieczne jest odtworzenie wybranej wersji historycznej danego pliku, należy wykonać operację `update` ze wskazaniem rewizji którą chcemy uzyskać. Sposób przekazania parametru dla polecenia `update` jest zależny od wybranego programu.

## 3.2 Narzędzia dostępu do repozytorium w trybie pełnego wersjonowania (SVN)

W tej sekcji przedstawione zostaną następujące programy:

- SmartSVN (p. [3.2.1](#))
- TortoiseSVN (p. [3.2.2](#))
- Polecenie `svn` (p. [3.2.3](#))

Do korzystania z repozytoriów SVN wystarczy **jeden** wybrany program. W tym dokumencie opisano korzystanie z różnych aplikacji by pozostawić użytkownikowi możliwość wyboru na wypadek gdyby np. z użyciem któregoś programu występowały problemy.

### 3.2.1 SmartSVN

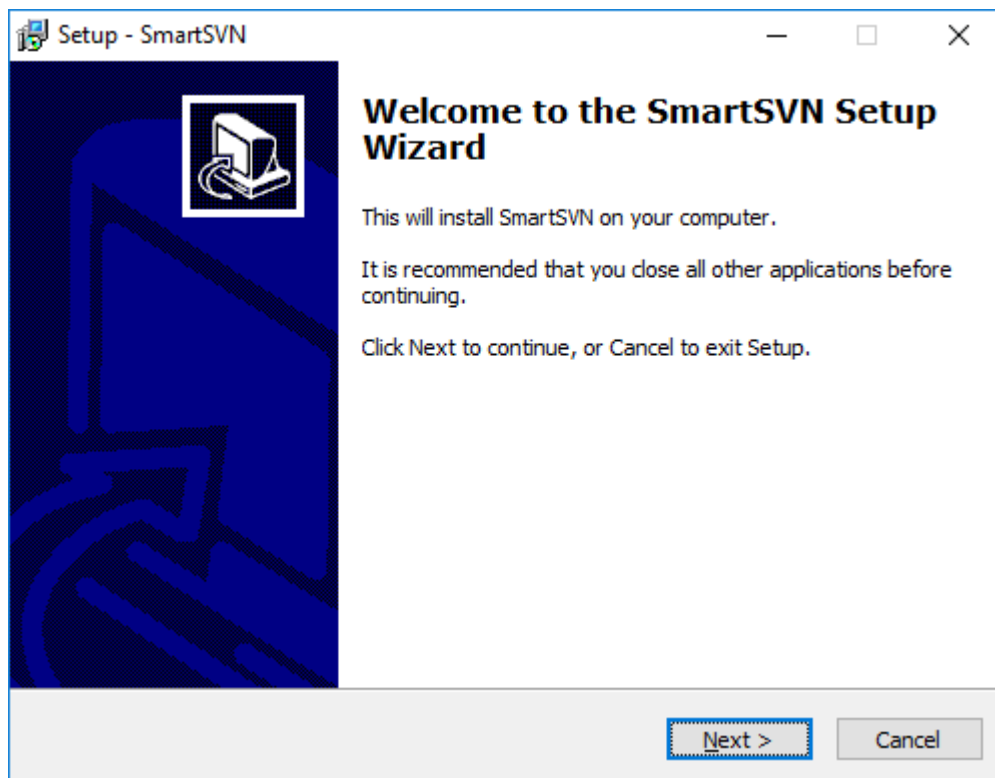
SmartSVN jest aplikacją do obsługi systemu Subversion napisaną w języku Java, dostępną dla systemów Windows, Linux i Mac OS X. SmartSVN jest bezpłatny w wersji Foundation, która w zupełności wystarczy do normalnej pracy.

## Zasoby

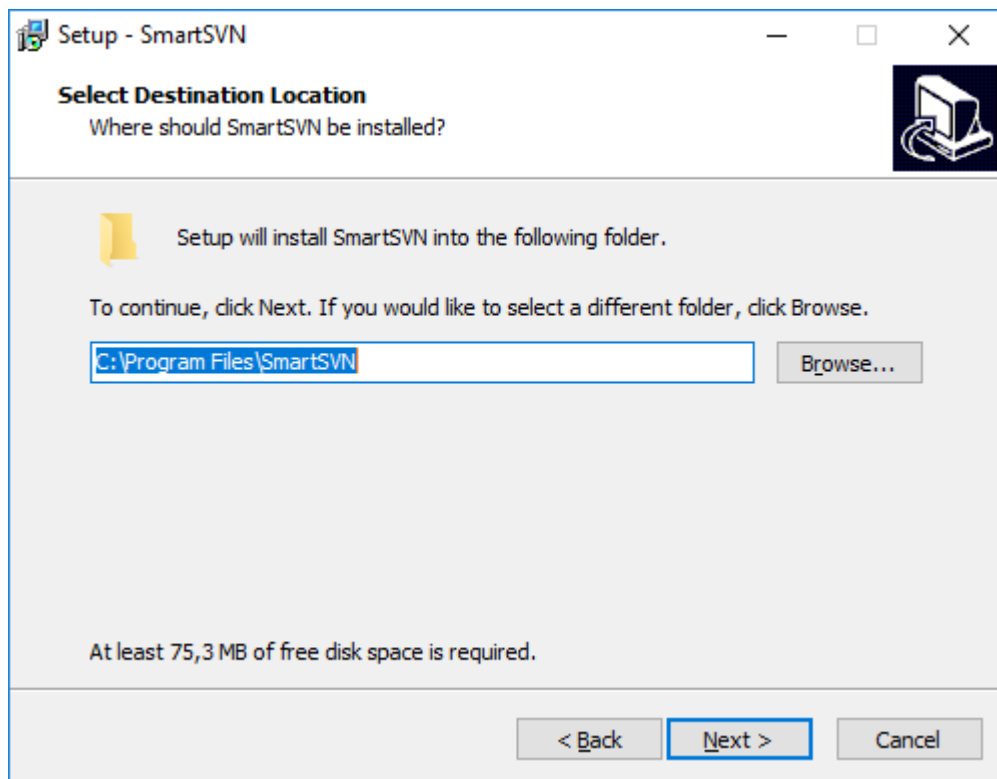
- Strona programu: <https://www.smartsvn.com/>
- [smartsvn-linux-11\\_0\\_0.tar.gz](#) - Wersja dla Linuxa
- [smartsvn-macosx-11\\_0\\_0.dmg](#) - Wersja dla Mac OS X
- [smartsvn-windows-portable-11\\_0\\_0.zip](#) - Wersja dla MS Windows (przenośna)
- [smartsvn-windows-setup-11\\_0\\_0.zip](#) - Wersja dla MS Windows (instalacyjna)

## Instalacja i uruchomienie

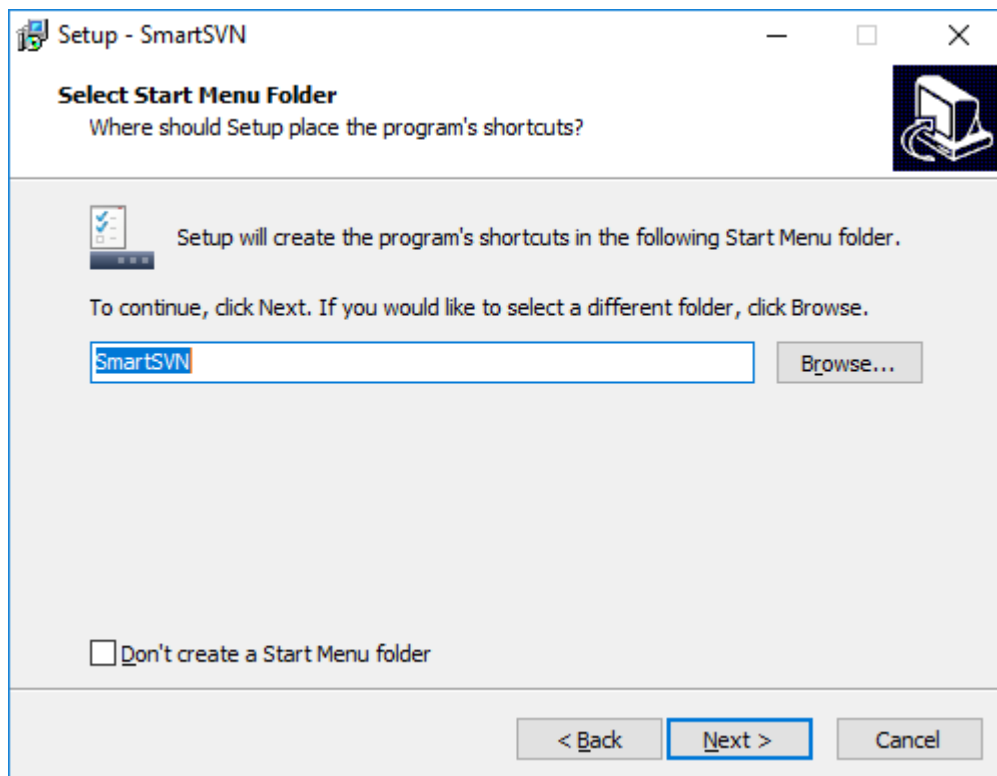
Proces instalacji i pierwszego uruchomienia jest przedstawiony poniżej:



Rysunek 17: Ekran powitalny programu instalacyjnego

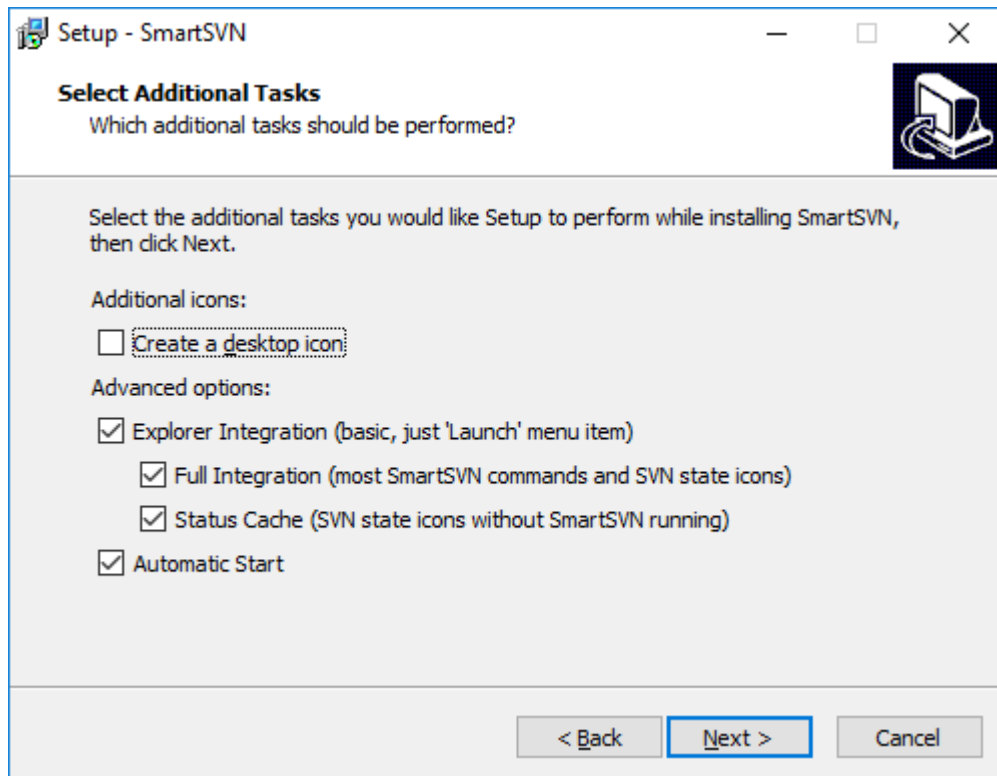


Rysunek 18: Wybór lokalizacji w której ma zostać zainstalowany program



Rysunek 19: Nazwa grupy programów w Menu Start

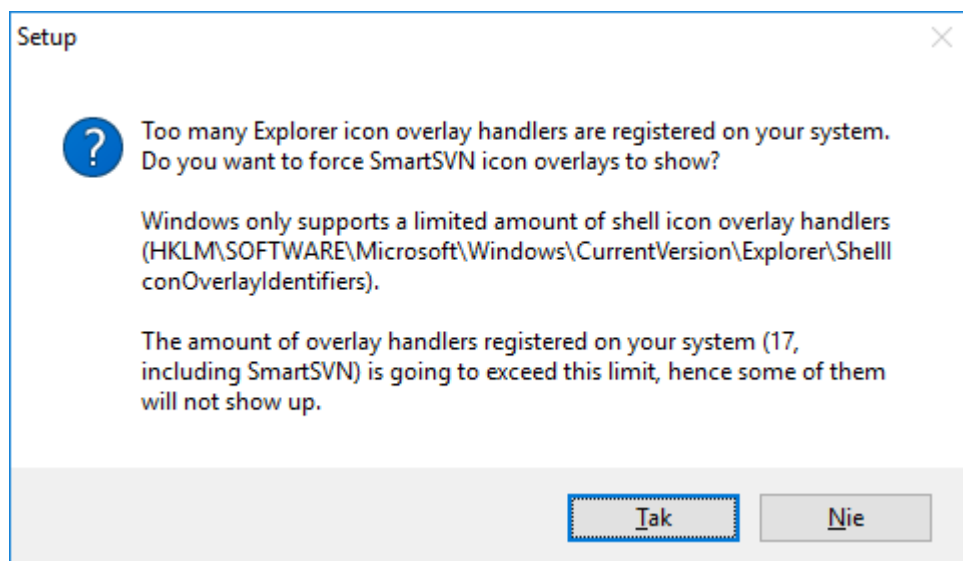
W czasie instalacji należy zdecydować czy program SmartSVN ma zostać zintegrowany z powłoką systemu Windows, dostarczając funkcjonalność zbliżoną do programu TortoiseSVN (3.2.2), która umożliwia wykonywanie operacji związanych z wersjonowaniem plików z menu kontekstowego Eksploratora Windows (kliknięcie prawym guzikiem myszy na wybrany plik lub katalog).



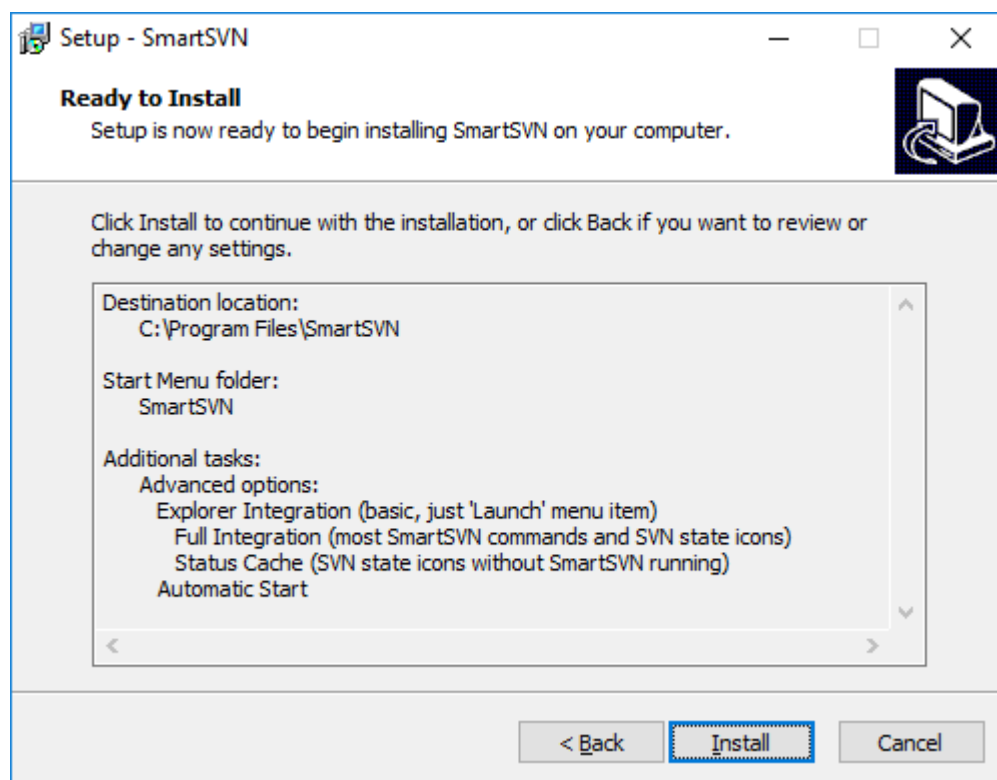
Rysunek 20: Opcje instalacji programu SmartSVN

W przypadku wybrania integracji z powłoką systemu Windows, może wyskoczyć błąd pokazany na rysunku 21. Jest on związany z wyświetlaniem ikon i nie ma wpływu na funkcjonowanie wersjonowania plików, więc można go zignorować.



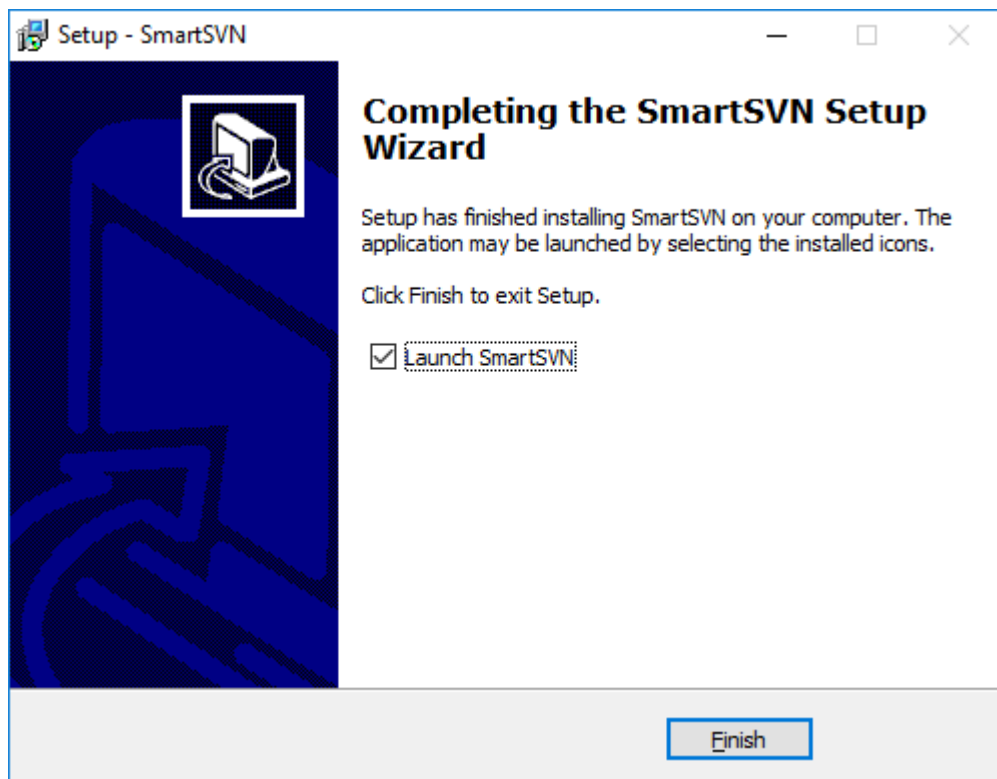


Rysunek 21: Błąd związany z wyświetlaniem ikon w systemie Windows



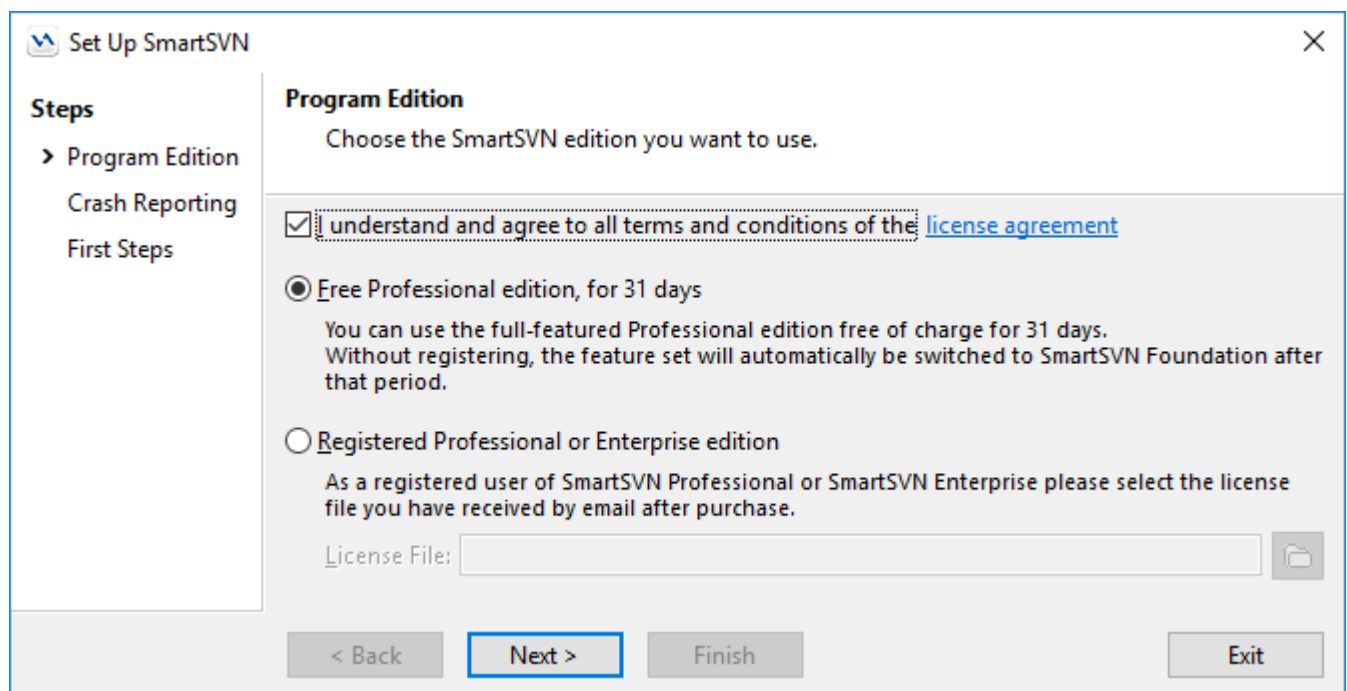
Rysunek 22: Program gotowy do instalacji

Po zainstalowaniu można od razu uruchomić program SmartSVN



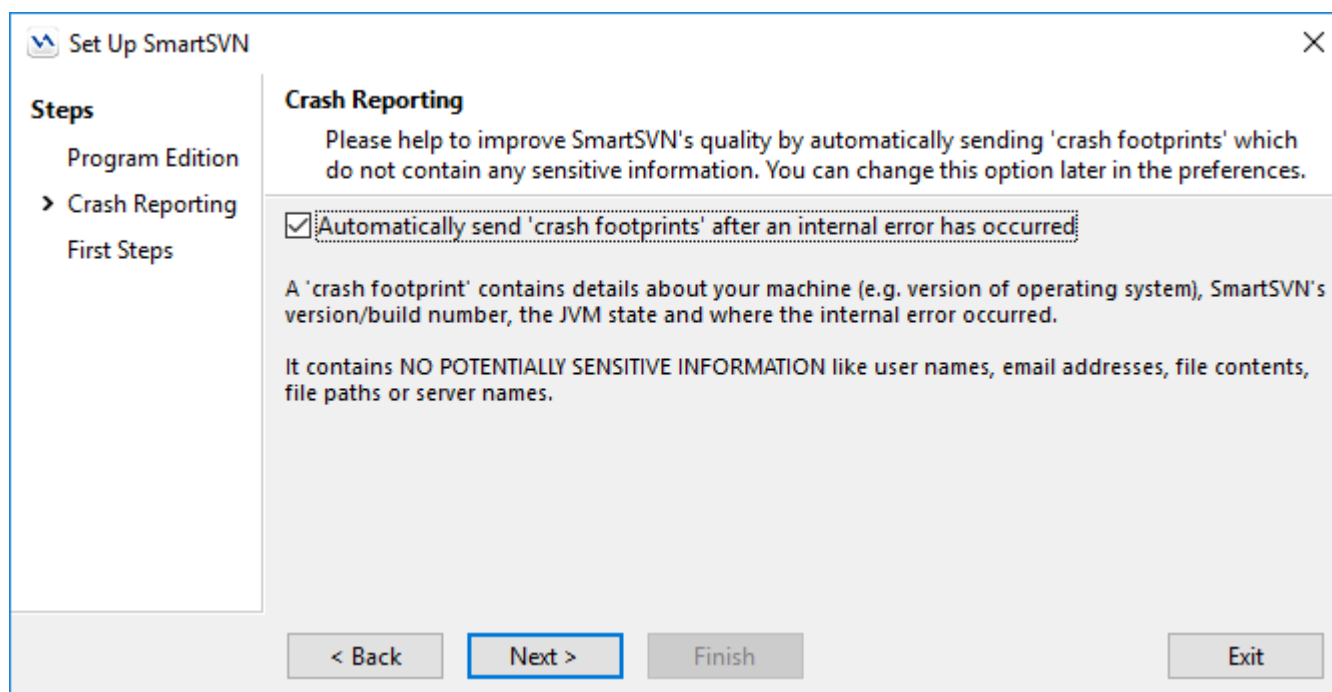
Rysunek 23: Program SmartSVN po zakończeniu instalacji

Po pierwszym uruchomieniu programu SmartSVN, należy wybrać rodzaj licencji (rys. 24). Bezpłatna licencja typu *Foundation*, do której zostanie ograniczona po 31 dniach domyślna licencja *Professional*, jest w zupełności wystarczająca do normalnej pracy.



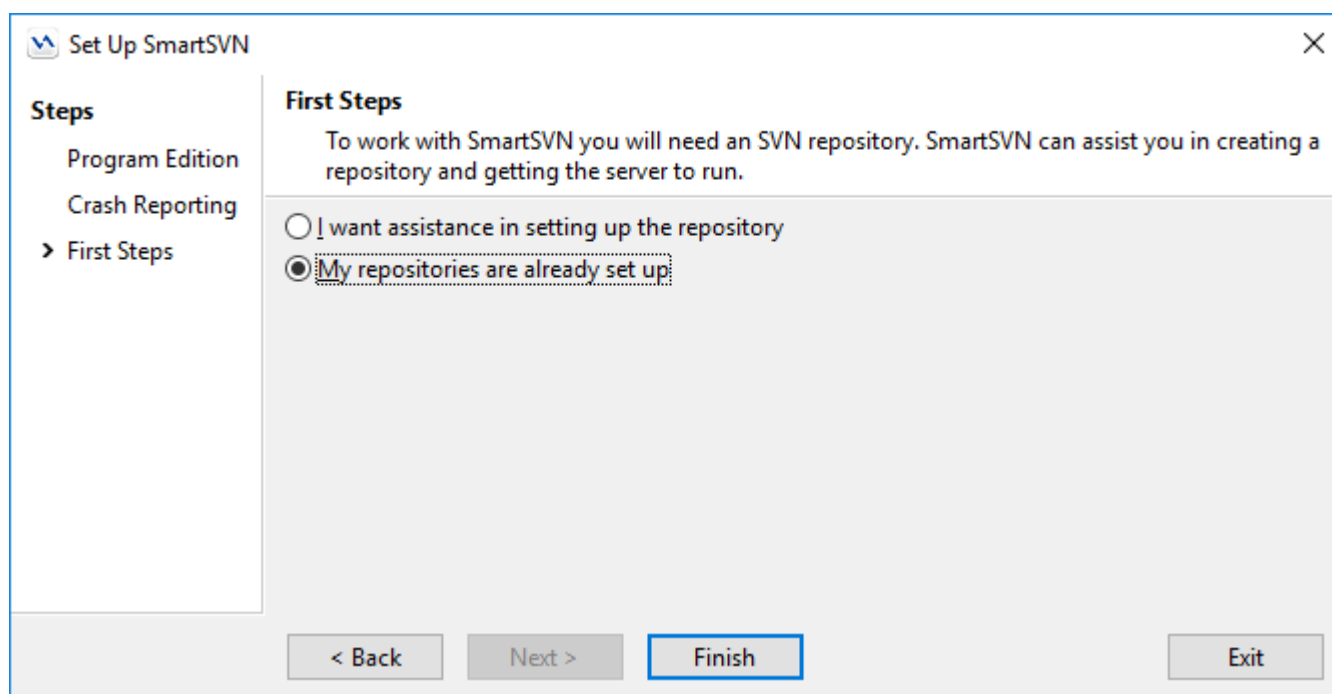
Rysunek 24: Pierwsze uruchomienie programu SmartSVN

Wysyłanie raportów o awariach programu SmartSVN do jego twórców (rys. 25) jest dobrowolne (wg. uznania użytkownika) i nie ma wpływu na funkcjonowanie z programu.



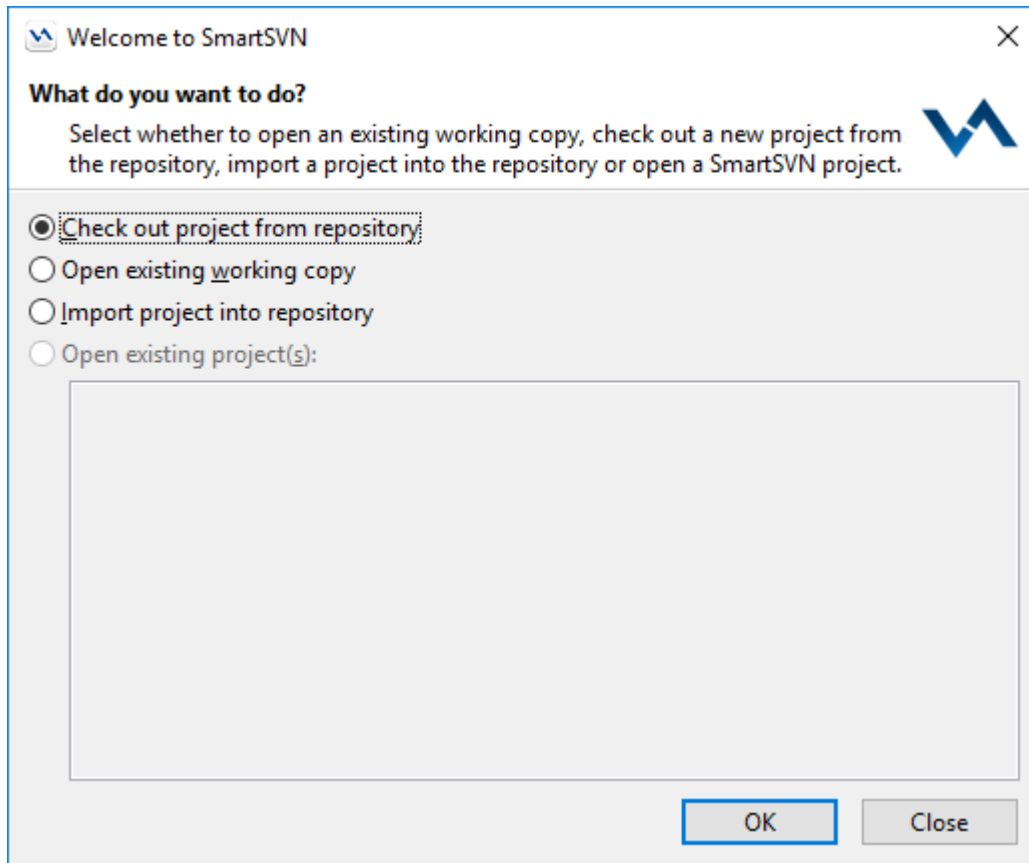
Rysunek 25: Opcje przesyłania raportów o awariach programu do jego twórców

W celu korzystania z programu do kontroli wersji, niezbędne jest posiadanie repozytoriów z plikami. W opisywanym przypadku repozytoria zostały już utworzone i taką opcję należy wybrać w czasie konfiguracji programu SmartSVN (rys. 26).



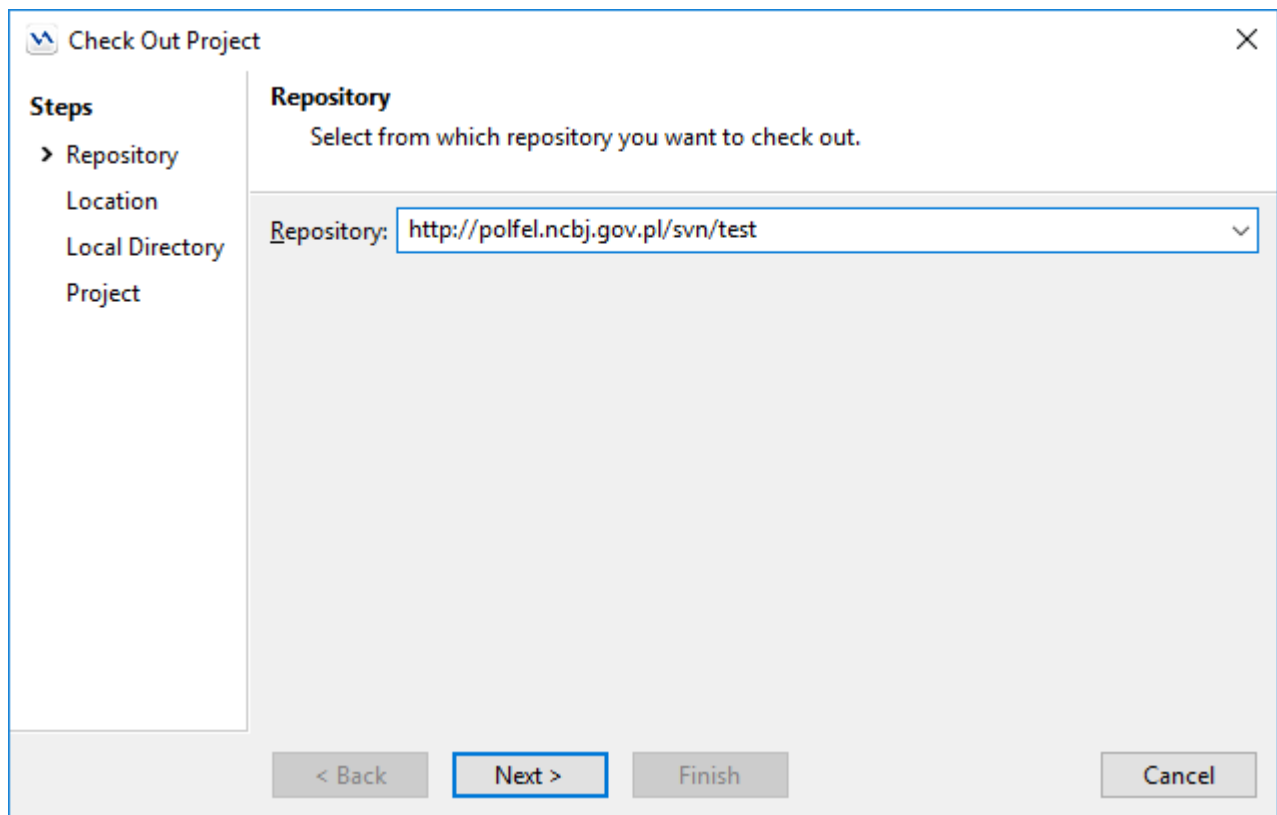
Rysunek 26: Konfiguracja repozytoriów w czasie pierwszego uruchomienia programu SmartSVN

W celu rozpoczęcia pracy, należy pobrać z wybranego repozytorium pliki, tworząc lokalnie kopię roboczą. Należy tego dokonać wykonując operację *checkout* (rys. 27).



Rysunek 27: Możliwe scenariusze rozpoczęcia pracy z programem SmartSVN

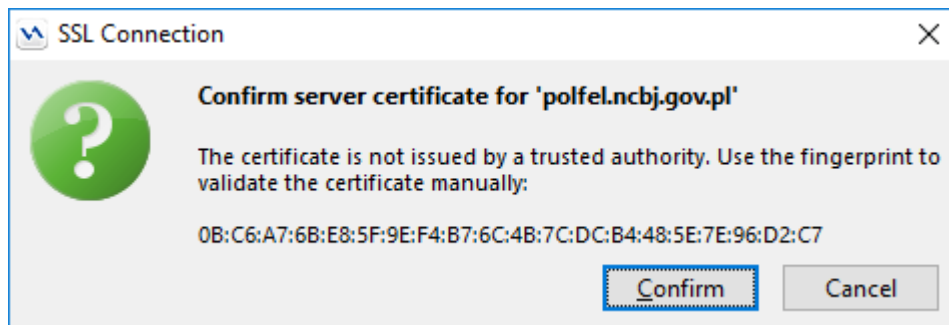
W celu przeprowadzenia operacji *checkout* należy podać adres wybranego repozytorium (rys. 28).



Rysunek 28: Wybór adresu repozytorium

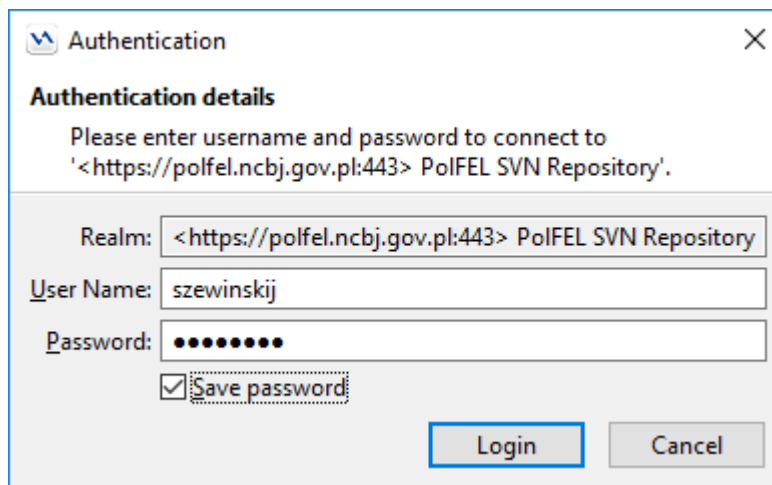
Z uwagi iż połączenie z repozytorium jest szyfrowane protokołem SSL, przy pierwszym połączeniu z danym serwerem niezbędne jest zaakceptowanie certyfikatu SSL dla tego serwera (rys. 29). Przy podłączaniu kolejnych repozytoriów z tego samego serwera, akceptowanie certyfikatu SSL nie powinno występować.

Spontaniczne wystąpienie (bez jasnej przyczyny) komunikatu jak na rys. 29, gdy wcześniej wielokrotnie korzystano z repozytoriów na danym serwerze, może oznaczać próbę ataku typu *man in the middle*. W takiej sytuacji nie należy ponownie podawać swoich danych autoryzacyjnych (loginu i hasła), lecz należy się skontaktować z osobą odpowiedzialną za wybrany projekt.



Rysunek 29: Akceptacja certyfikatu SSL serwera

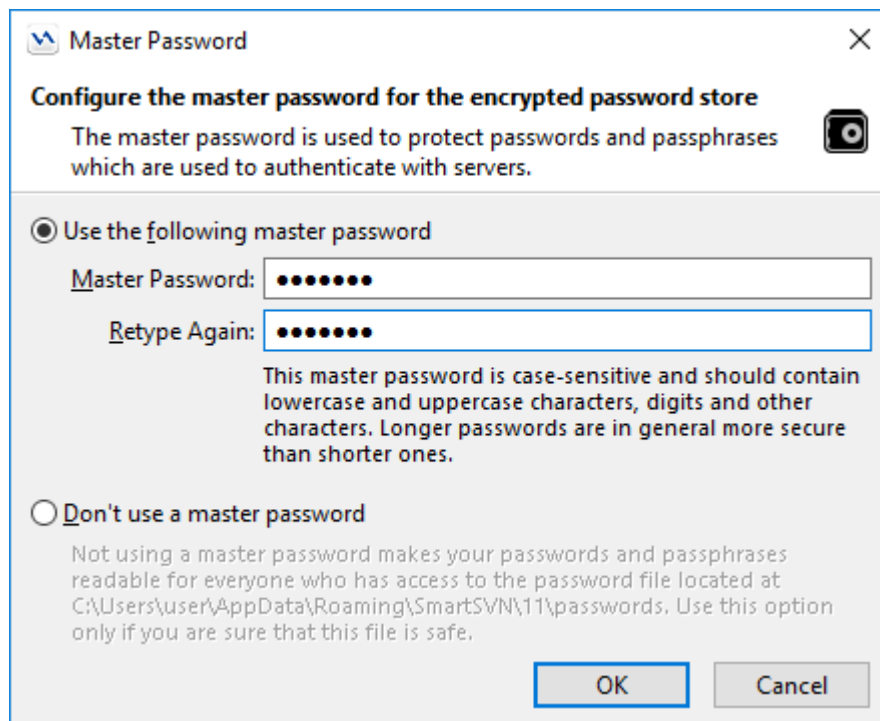
Po wybraniu repozytorium (i ew. zaakceptowaniu certyfikatu SSL jeśli to pierwsze połączenie z danym serwerem), należy podać swój login oraz hasło (rys. 30).



Rysunek 30: Autoryzacja dostępu do wybranego repozytorium

W celu zachowania najwyższego poziomu bezpieczeństwa, program SmartSVN oferuje możliwość zaszyfrowania swojej lokalnej bazy/listy haseł dostępu do obsługiwanych repozytoriów. Zaszyfrowanie lokalnej bazy haseł wymaga ustawienia hasła nadrzędnego (ang. *master password*, rys. 31).

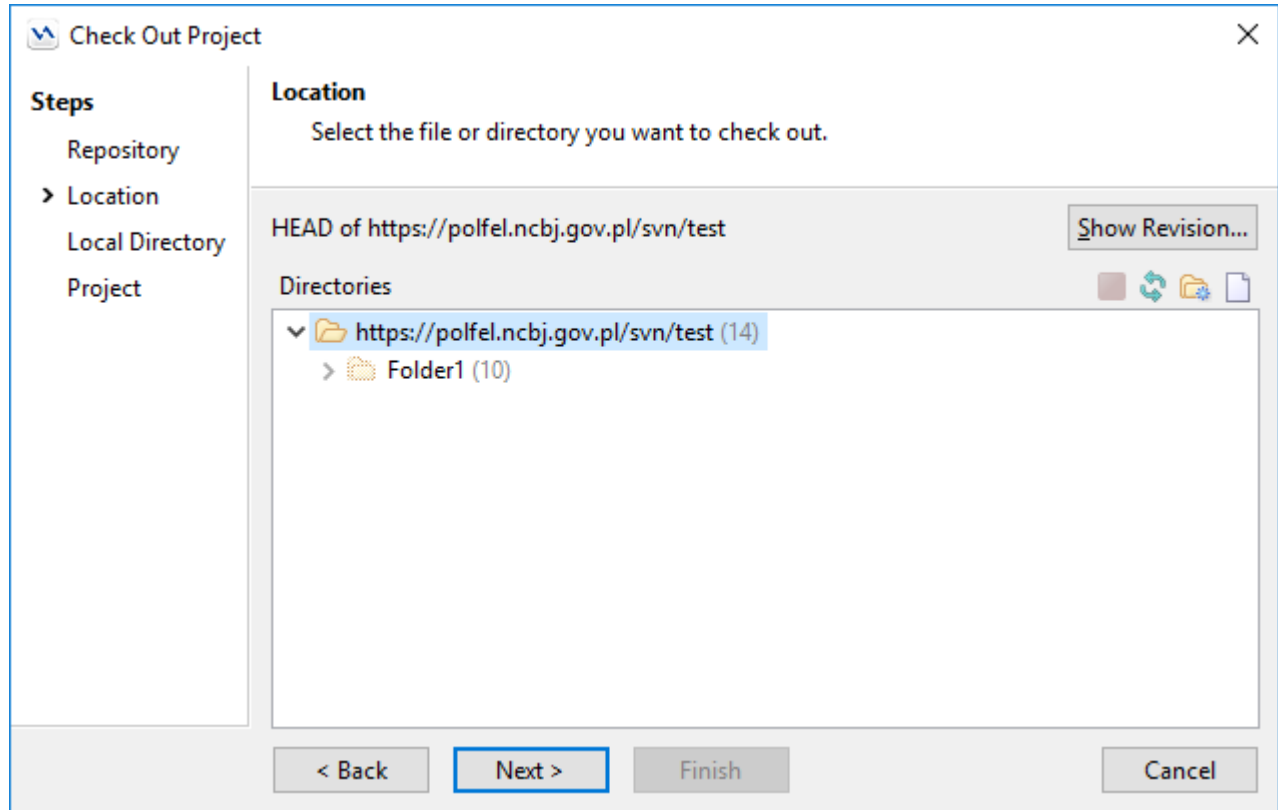
Pomimo iż nie ma to wpływu na funkcjonowanie programu, zaszyfrowanie bazy haseł jest zalecane.



Rysunek 31: Możliwość zaszyfrowania lokalnej bazy haseł hasłem nadrzędnym

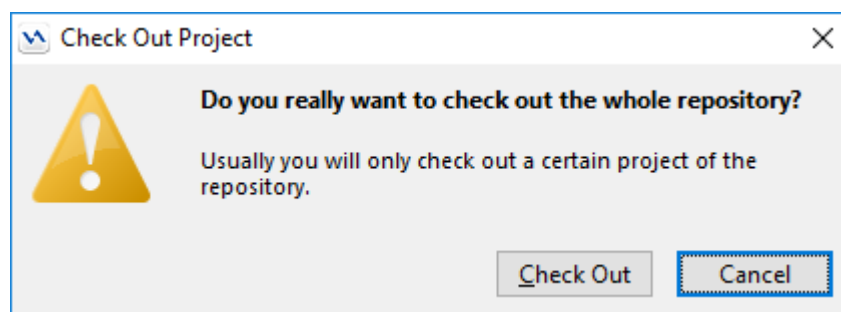
Po podaniu poprawnego adresu repozytorium, loginu i hasła, możliwe jest nawiązanie połączenia i wyświetlanie zawartości repozytorium oraz wybór zakresu plików do pobrania (rys. 32).

Subversion pozwala zarówno na utworzenie kopii roboczej w oparciu całe repozytorium lub o wybrany pod-folder.



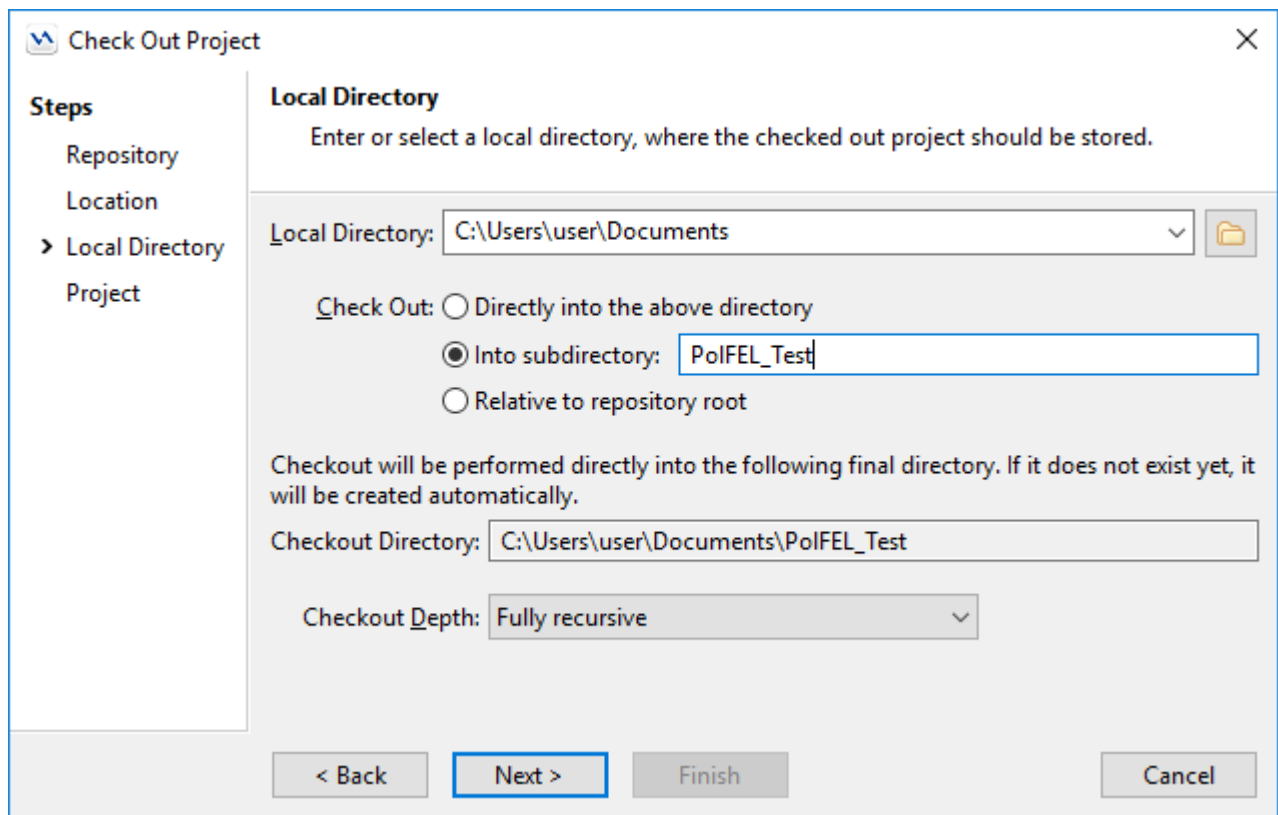
Rysunek 32: Wyświetlenie zawartości repozytorium, wybór zakresu danych do pobrania

W przypadku próby pobrania całego repozytorium, program SmartSVN pyta o potwierdzenie (rys. 33), gdyż w przypadku dużych repozytoriów mogłoby to zająć dużo czasu i miejsca na dysku



Rysunek 33: Potwierdzenie pobrania całego repozytorium

Po wybraniu zakresu plików w repozytorium do pobrania, należy określić w którym miejscu na dysku ma zostać utworzona kopia robocza (rys. 34).

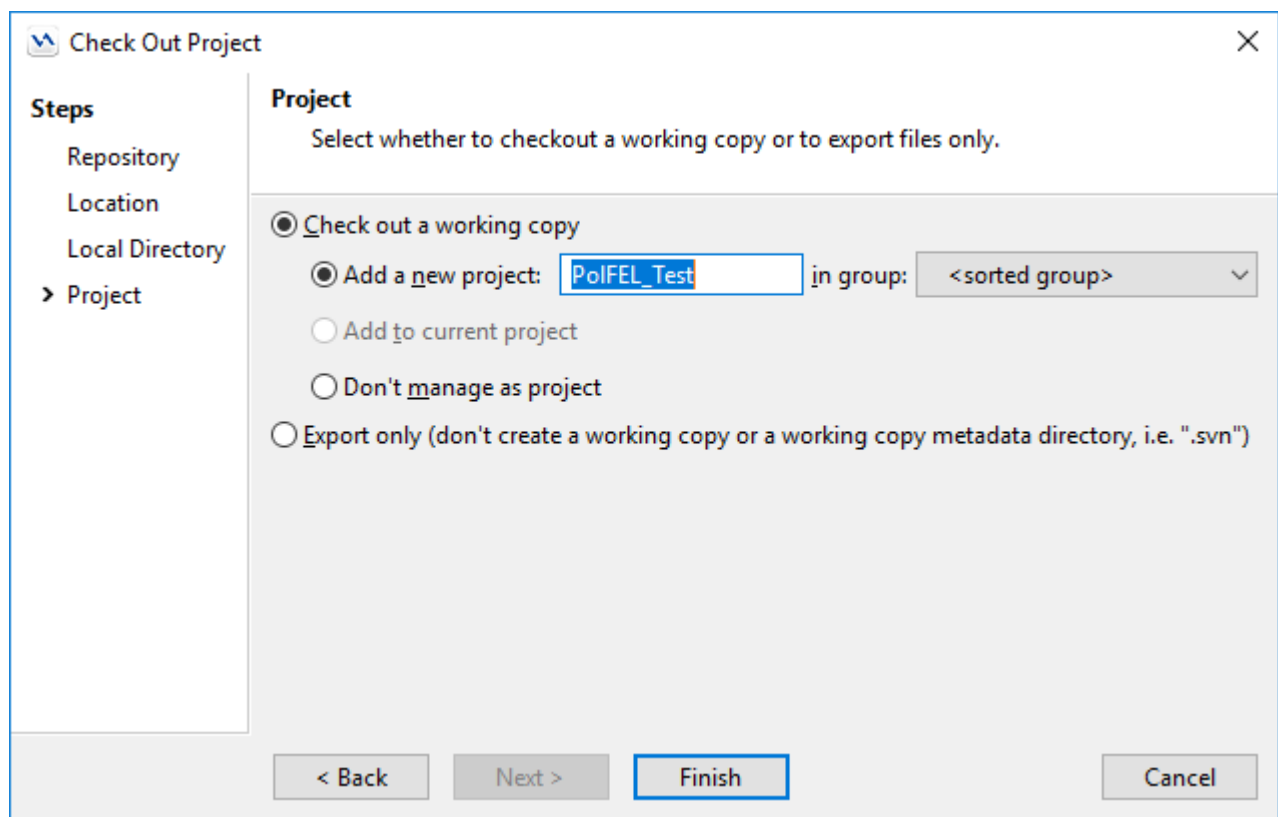


Rysunek 34: Określenie lokalizacji utworzenia kopii roboczej

Ostatnim etapem operacji *checkout* jest ew. utworzenie projektu (rys. 35). *Projekt* jest pojęciem wewnętrznym programu SmartSVN, nie jest on częścią standardu Subversion. *Projekt* umożliwia on dalszą obsługę utworzonej kopii roboczej przez program SmartSVN.

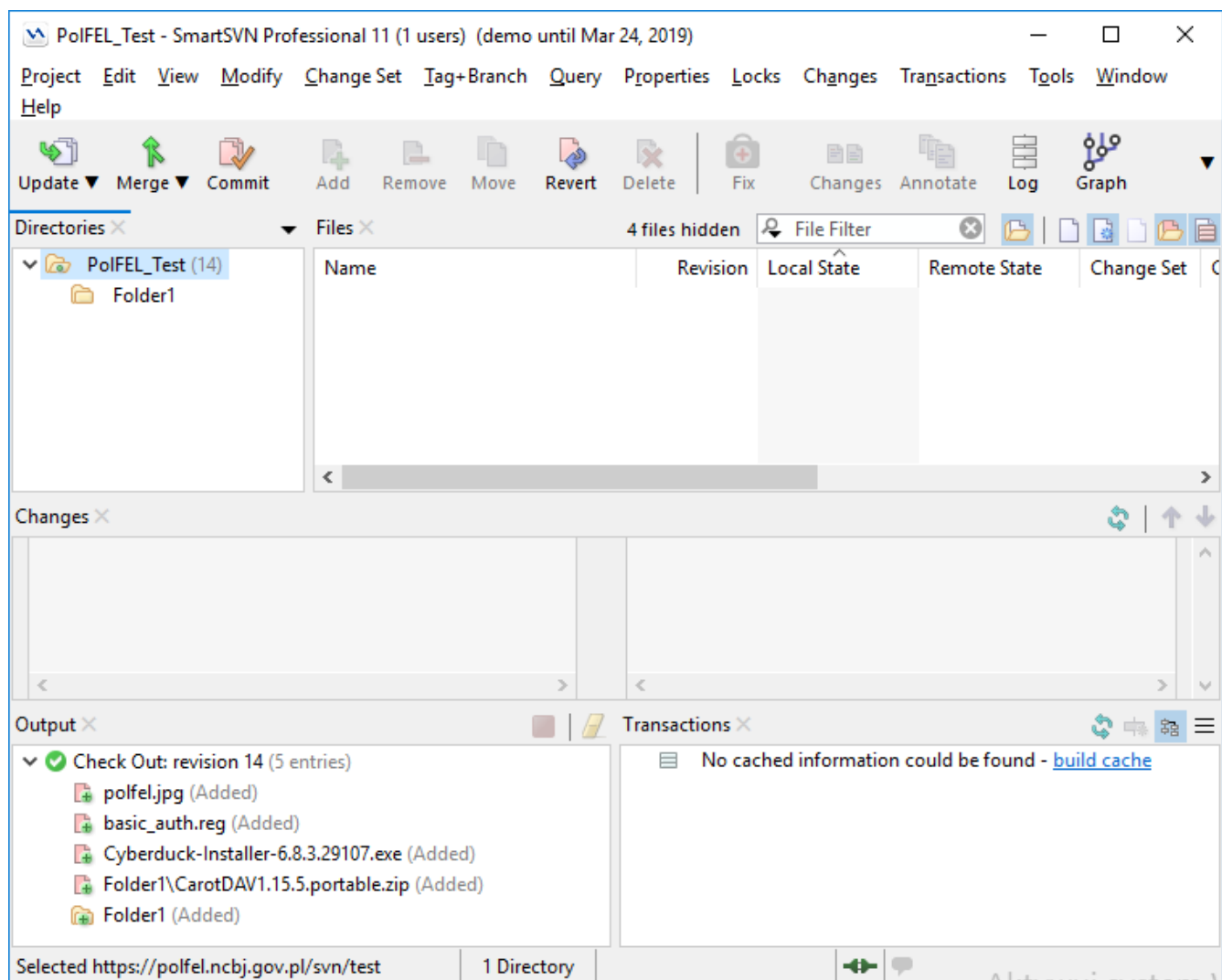
Utworzenie projektu jest zalecane w celu normalnego korzystania z programu SmartSVN.





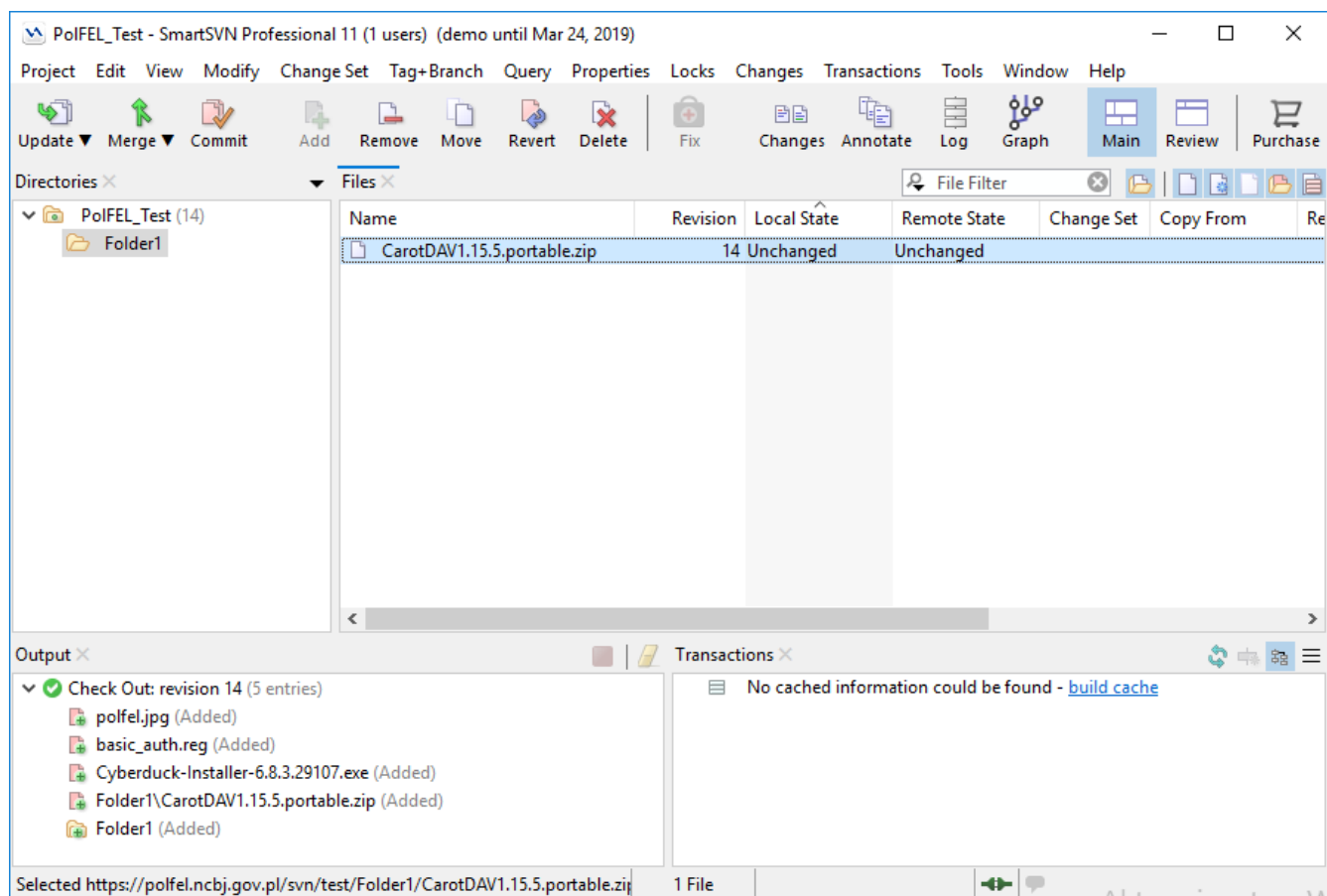
Rysunek 35: Opcje dalszej obsługi nowopowstałej kopii roboczej

Po utworzeniu i otwarciu projektu powiązanego z wybraną kopią roboczą możliwa jest praca z wersjonowanymi plikami (rys. 36)



Rysunek 36: Okno programu SmartSVN z otwartą kopią roboczą

Po wybraniu odpowiedniego pliku (zależnie od jego stanu), możliwe jest wykonywanie wybranych operacji SVN, takich jak np. *update*, *commit*, *revert*, itp. (rys. 37)



Rysunek 37: Dostępne opcje programu po wybraniu danego pliku

### 3.2.2 TortoiseSVN

TortoiseSVN jest klientem Subversion który integruje się z powłoką systemu operacyjnego MS Windows, i umożliwia wykonywanie operacji związanych z wersjonowaniem plików z poziomu menu kontekstowego (po kliknięciu prawym guzikiem myszy) w wybrany plik lub katalog.

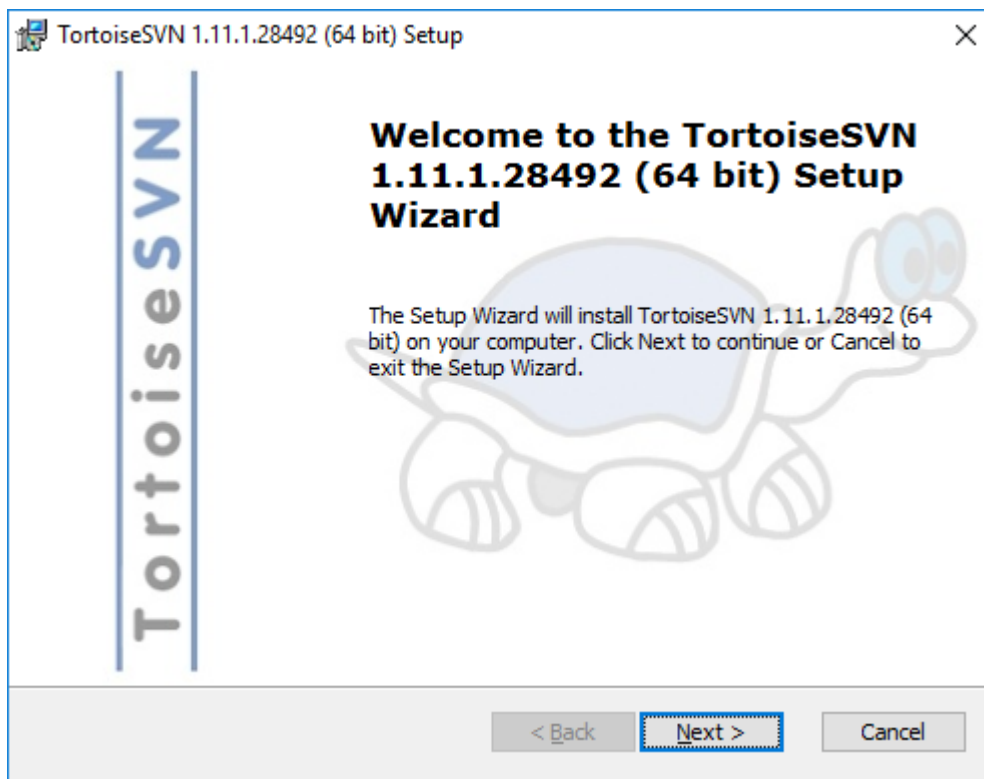
Odpowiednikiem TortoiseSVN dla platformy Mac OS X jest [SnailSVN](#)

#### Zasoby

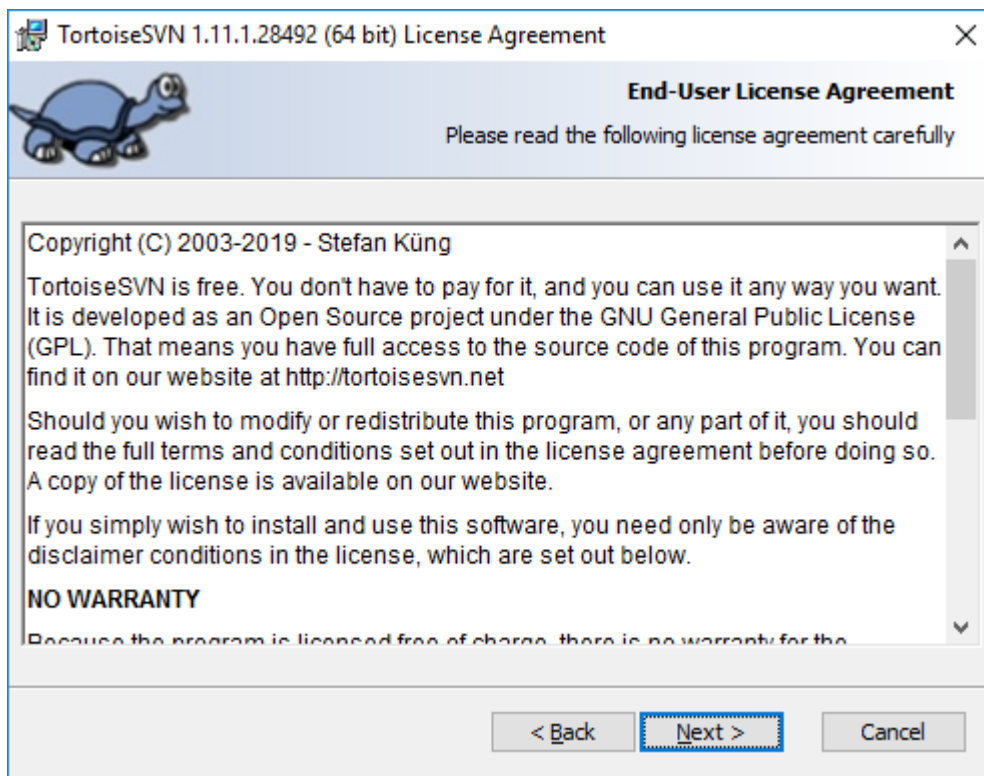
- Strona programu: <https://www.smartsvn.com/>
- [TortoiseSVN-1.11.1.28492-win32-svn-1.11.1.msi](#) Wersja instalacyjna 32-bitowa
- [TortoiseSVN-1.11.1.28492-x64-svn-1.11.1.msi](#) Wersja instalacyjna 64-bitowa

#### Instalacja i uruchomienie

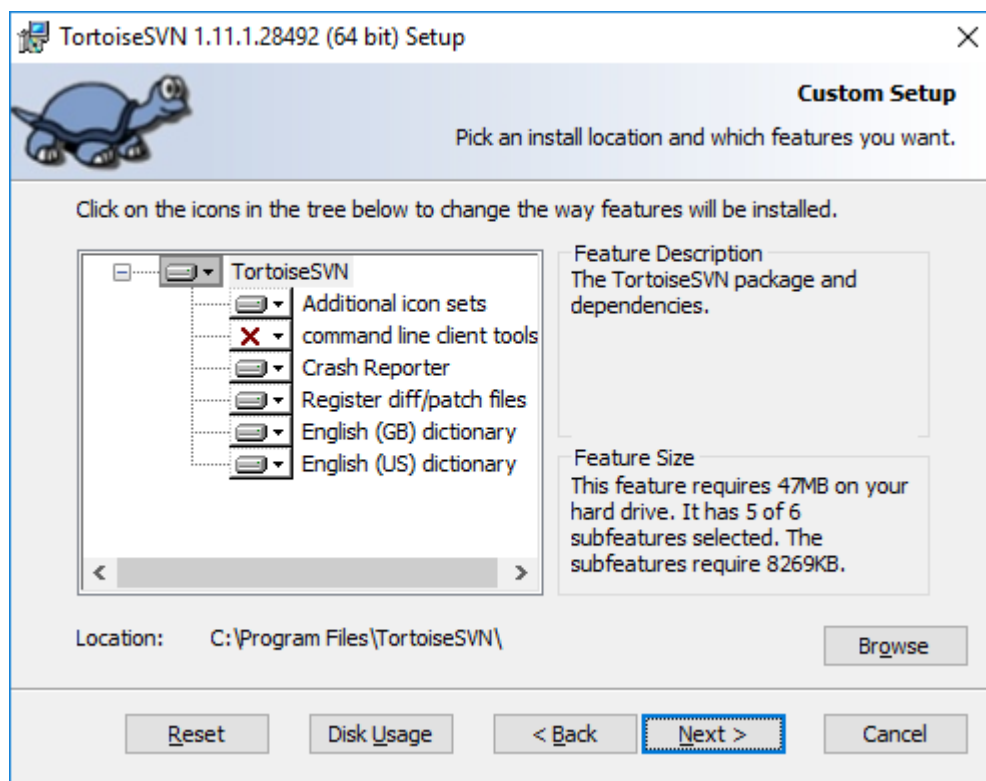
Proces instalacji i uruchomienia jest przedstawiony poniżej:



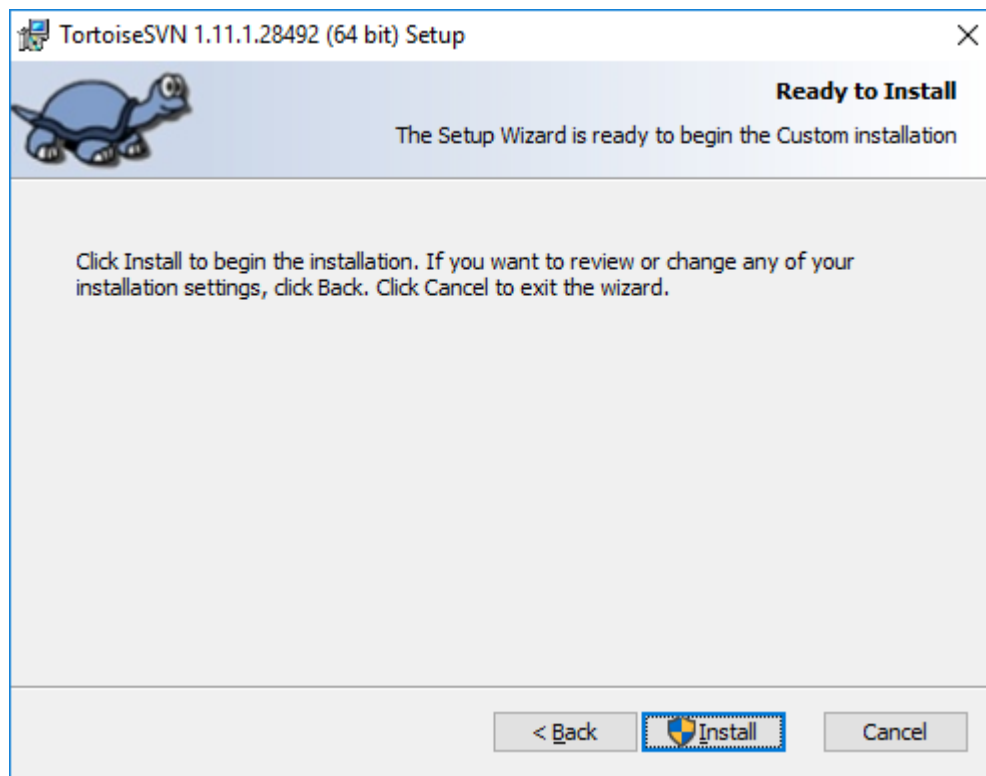
Rysunek 38: Ekran powitalny programu instalacyjnego TortoiseSVN



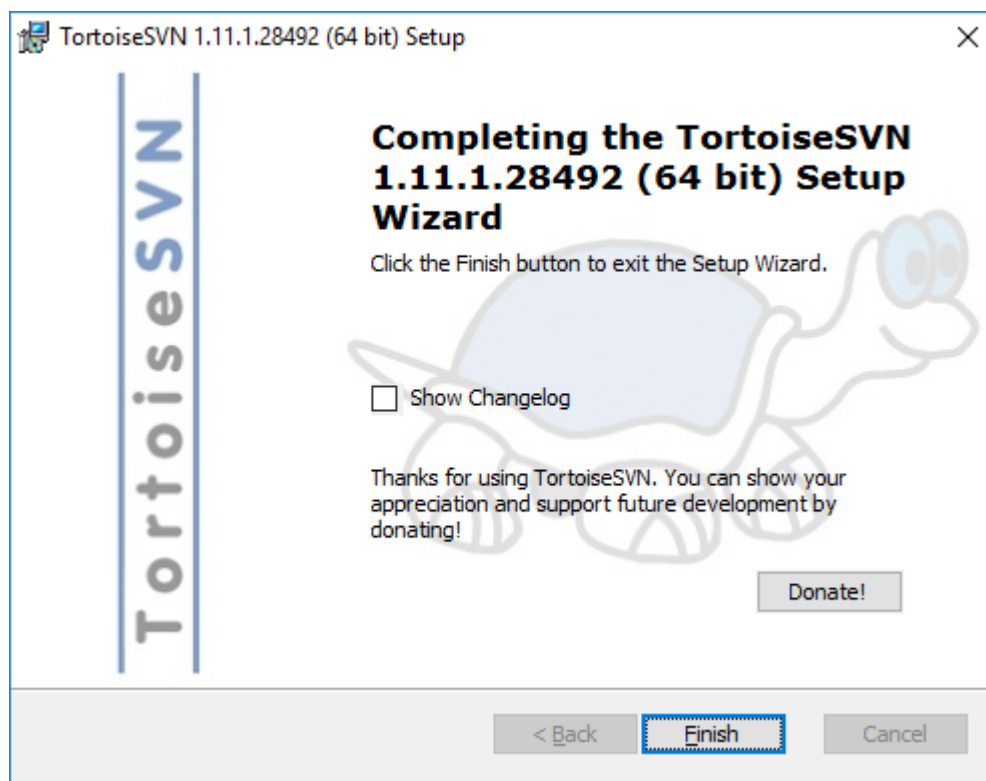
Rysunek 39: Licencja programu TortoiseSVN



Rysunek 40: Opcje instalacji programu TortoiseSVN

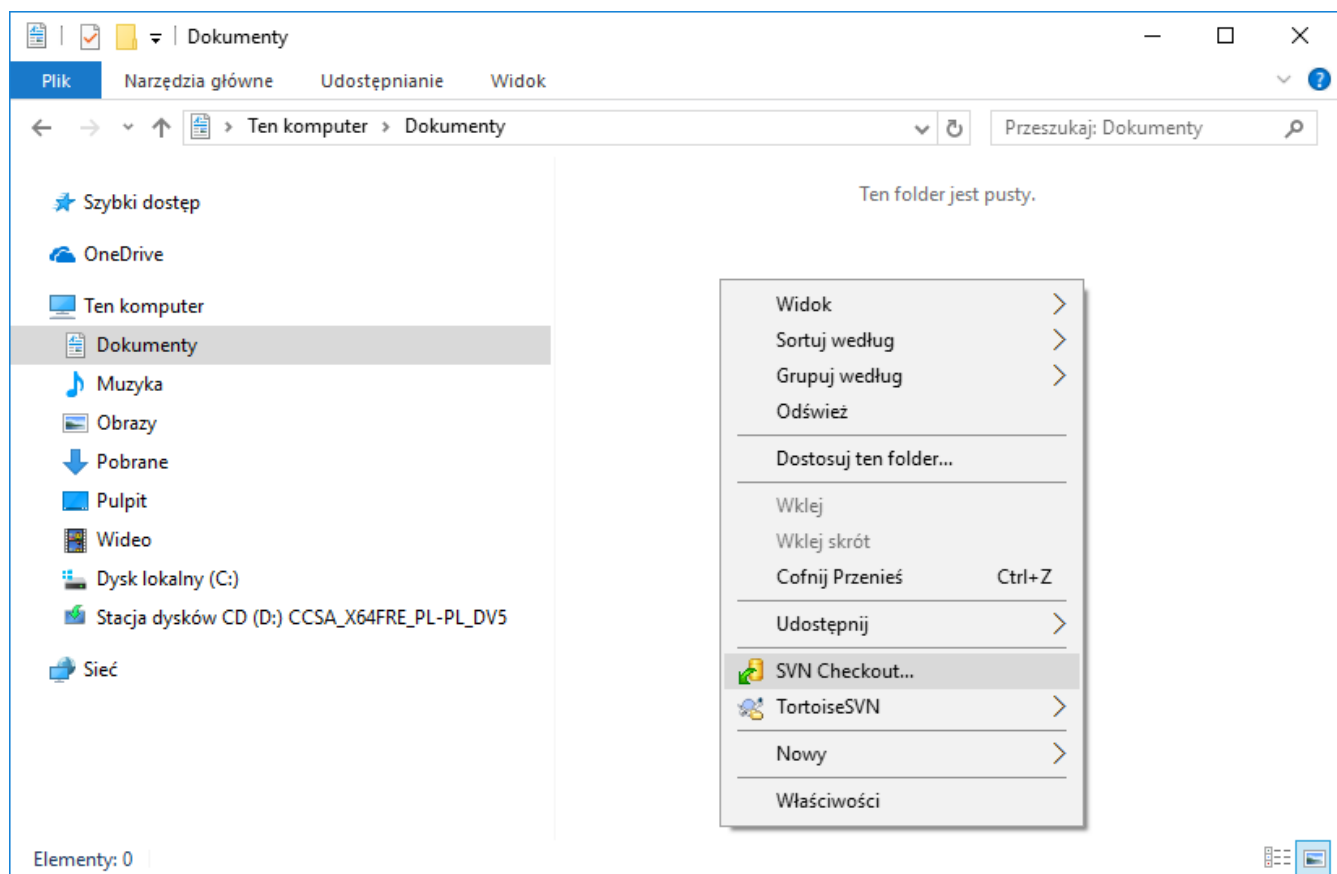


Rysunek 41: Program gotowy do instalacji



Rysunek 42: Program po instalacji

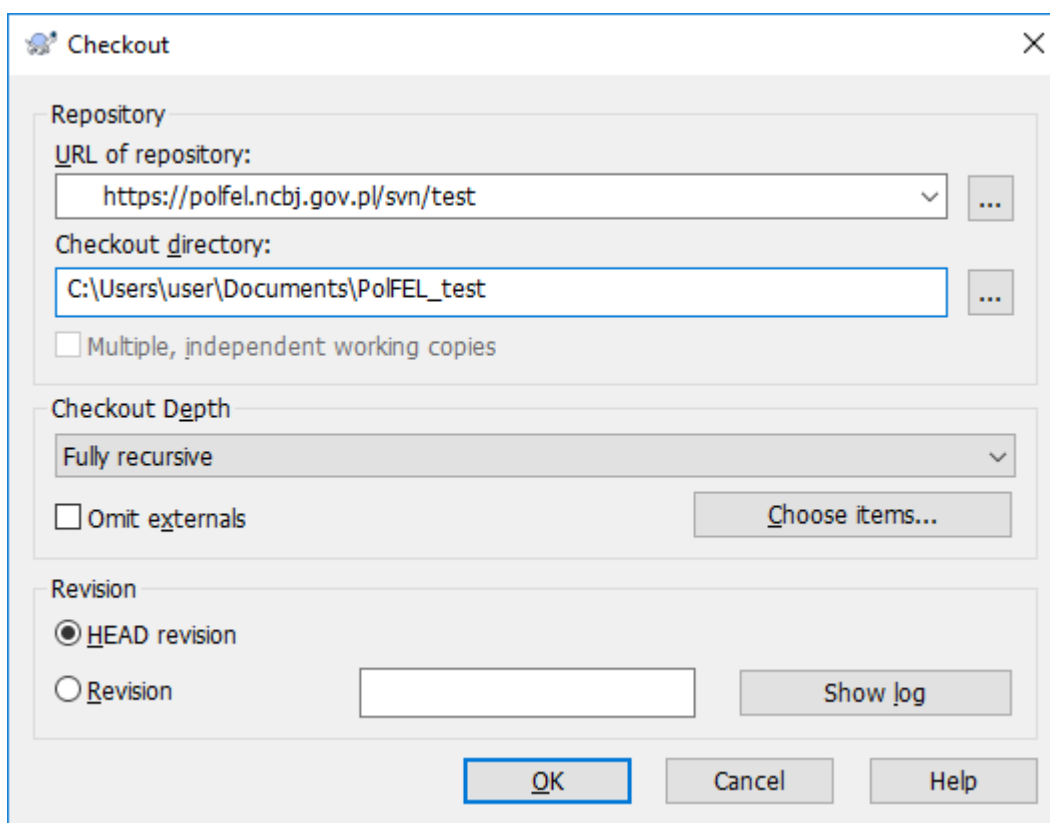
W celu utworzenia kopii roboczej, należy kliknąć prawym guzikiem myszy w tło danego folderu i wybrać opcję *SVN Checkout* (rys. 43).



Rysunek 43: Tworzenie kopii roboczej w wybranym folderze systemu MS Windows

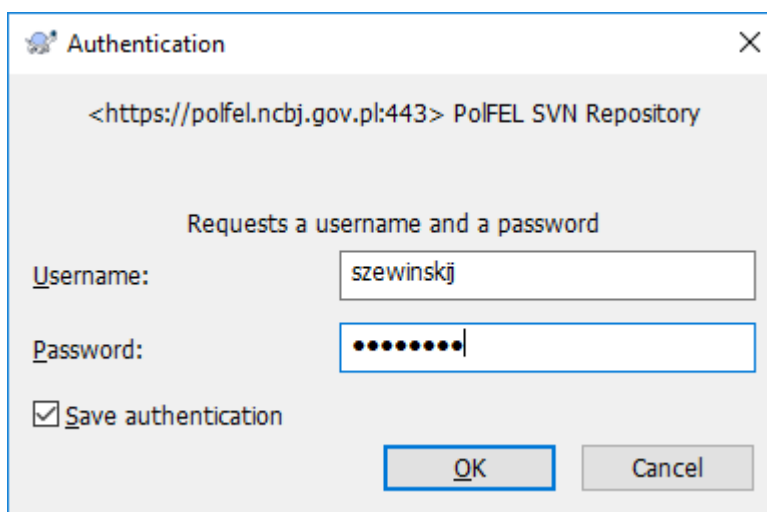
By utworzyć kopię roboczą, należy podać m.in. (rys. 44):

- URL (adres) Repozytorium
- Katalog w którym ma powstać nowa kopia robocza
- Rewizję którą chcemy pobrać (*HEAD* oznacza najbardziej aktualną rewizję).



Rysunek 44: Opcje operacji *checkout*

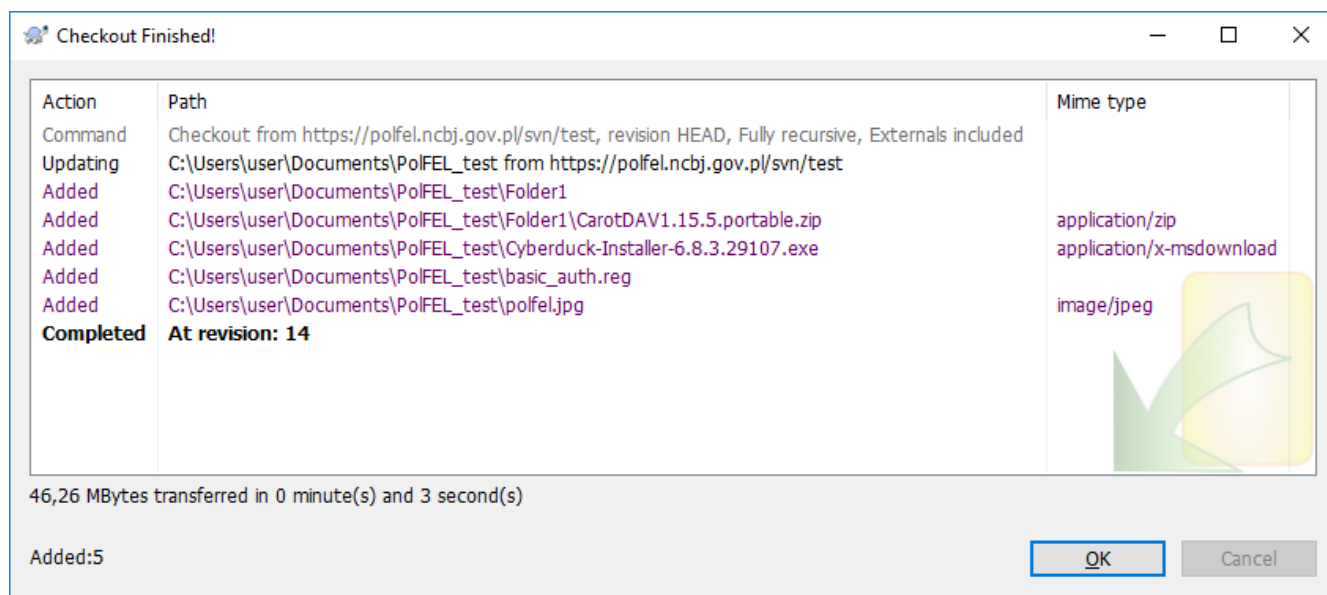
By uzyskać dostęp do repozytorium, niezbędne jest podanie danych autoryzacyjnych - loginu i hasła (rys. 45).



Rysunek 45: Autoryzacja dostępu do repozytorium

Po poprawnym przeprowadzeniu operacji *checkout*, wyświetlone zostanie podsumowanie (rys. 46).

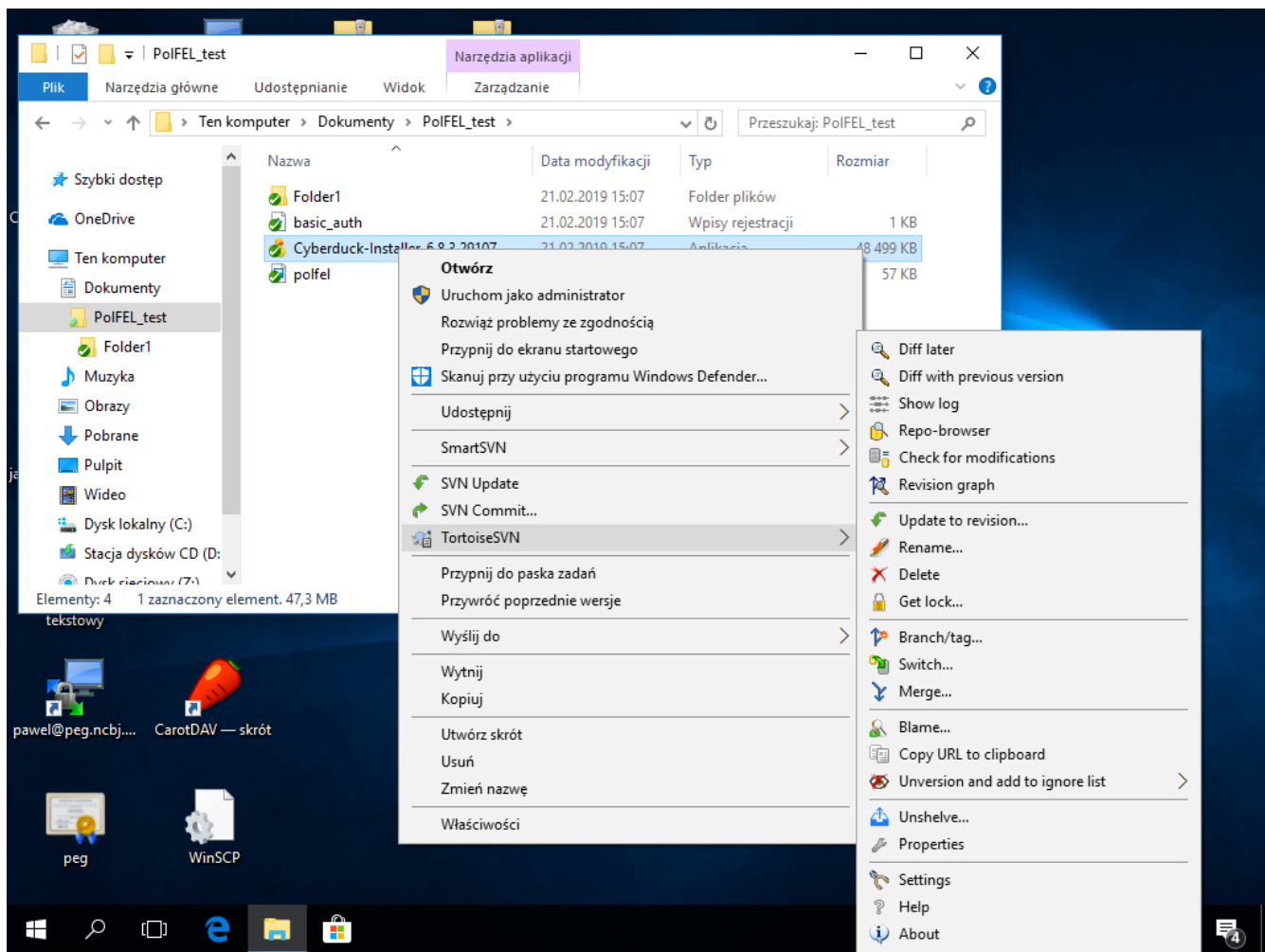




Rysunek 46: Podsumowanie poprawnie przeprowadzonej operacji *checkout*

W celu wykonania wybranych operacji SVN, należy kliknąć na wybrany plik lub katalog prawym guzikiem myszy, i z menu *TortoiseSVN* wybrać daną operację SVN (47).

Zależnie od stanu pliku, różne operacje mogą być dostępne, dla nowego (niewersjonowanego) pliku dostępna będzie operacja *add*, dla wersjonowanego pliku dostępna będzie operacja *update*, itp.



Rysunek 47: Dostępne operacje SVN dla wybranego pliku

### 3.2.3 Polecenie `svn` w systemach operacyjnych typu POSIX

W większości systemów operacyjnych typu POSIX, takich jak np. Linux, dostępne jest polecenie `svn`. Jest to podstawowa implementacja programu do obsługi systemu wersjonowania Subversion. Polecenie `svn` umożliwia wykonanie każdej operacji dostępnej w systemie Subversion. Program ten nie posiada graficznego interfejsu użytkownika, poszczególne operacje związane z wersjonowaniem plików należy wykonywać wydając odpowiednie komendy w wierszu poleceń.

Komendy mają ogólną strukturę:

```
svn <komenda> [opcjonalne parametry]
```

Lista dostępnych komend może być wyświetlona za pomocą:

```
svn help
```

szczegóły wybranej komendy można uzyskać wydając polecenie:

```
svn help <komenda>
```

## Przykłady użycia polecenia svn

Operacja Checkout:

```
jarek@carbon:~/polfel$ svn checkout https://polfel.ncbj.gov.pl/svn/test
Obszar uwierzytelniania: <https://polfel.ncbj.gov.pl:443> PolFEL SVN Repository
Hasło 'jarek':
```

```
Obszar uwierzytelniania: <https://polfel.ncbj.gov.pl:443> PolFEL SVN Repository
Użytkownik: szewinskij
Hasło 'szewinskij': *****
```

```
A    test/Folder1
A    test/Folder1/CarotDAV1.15.5.portable.zip
A    test/System Volume Information
A    test/System Volume Information/IndexerVolumeGuid
A    test/Cyberduck-Installer-6.8.3.29107.exe
A    test/basic_auth.reg
A    test/polfel.jpg
Pobrano wersję 29.
jarek@carbon:~/polfel$
```

Utworzenie nowego pliku i nazwie "plik.txt" i zawartości "test", oraz sprawdzenie stanu kopii roboczej - nowy plik "plik.txt" zostanie wskazany jako nieznanym (niewersjonowany):

```
jarek@carbon:~/polfel/test$ echo test > plik.txt
jarek@carbon:~/polfel/test$ svn status
?    plik.txt
```

Dodanie nowego pliku do kontroli wersji, plik zostanie wskazany jako dodany (A):

```
jarek@carbon:~/polfel/test$ svn add plik.txt
A    plik.txt
```

Zatwierdzenie dodania pliku i przesłanie zmian do repozytorium:

```
jarek@carbon:~/polfel/test$ svn commit -m "Dodano plik.txt"
Dodawanie      plik.txt
Przesyłanie treści pliku.done
Committing transaction...
Zatwierdzona wersja 30.
```

Zmiana zawartości pliku, plik zostanie wskazany jako zmodyfikowany (M):

```
jarek@carbon:~/polfel/test$ cat plik.txt
test
jarek@carbon:~/polfel/test$ echo "nowa zawartość" > plik.txt
jarek@carbon:~/polfel/test$ svn status
M    plik.txt
jarek@carbon:~/polfel/test$ cat plik.txt
nowa zawartość
```

Wycofanie lokalnej zmiany:

```
jarek@carbon:~/polfel/test$ cat plik.txt
nowa zawartość
jarek@carbon:~/polfel/test$ svn revert plik.txt
Wycofano zmiany w 'plik.txt'
jarek@carbon:~/polfel/test$ cat plik.txt
test
```

Usunięcie wersjonowanych plików z dysku oraz ich odtworzenie poleceniem update:

```
jarek@carbon:~/polfel/test$ rm basic_auth.reg plik.txt polfel.jpg
jarek@carbon:~/polfel/test$ svn status
!      basic_auth.reg
!      plik.txt
!      polfel.jpg
jarek@carbon:~/polfel/test$ svn update
Updating '.':
Odtworzono 'plik.txt'
Odtworzono 'basic_auth.reg'
Odtworzono 'polfel.jpg'
W wersji 30.
jarek@carbon:~/polfel/test$
```